CLIFFORD & MARK
RAMSHAW

COMMODORE 64

GAMES BOOK



INCLUDES UNIQUE CHEXSUM VERIFICATION PROGRAM

COMMODORE 64

GAMES BOOK

COMMODORE 64

GAMES BOOK

CLIFFORD & MARK
RAMSHAW

Melbourne House Publishers

Published in the United Kingdom by:
Melbourne House (Publishers) Ltd.,
Church Yard,
Tring, Hertfordshire HP23 5LU,
ISBN 0 86161 125 X

Published in Australia by:
Melbourne House (Australia) Pty. Ltd.,
Suite 4, Palmerston Crescent,
South Melbourne, Victoria 3205.

Published in the United States of America by:
Melbourne House Software Inc.,
347 Reedwood Drive,
Nashville TN 37217

PRINTED IN HONG KONG BY COLORCRAFT LTD.

D C B A 9 8 7 6 5 4 3 2 1

Dedicated to Mom and Dad

Publisher's Note

In keeping with our ongoing commitment to provide both literature and software for personal computers Melbourne House is very proud to be able to publish this book of games for the Commodore 64.

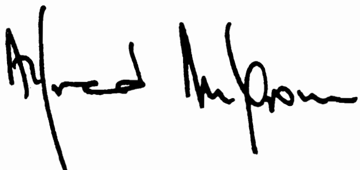
You will find complete program listings, comprehensive structures and useful hints on each program. We have taken care to design the format in such a way that the programs will be easily read, to reduce the possibility of transcription errors, expecially with the graphics characters.

After working through this book, I think you will agree that Clifford and Mark Ramshaw are immensely creative programmers and the programs in this book should set the standard future programs for the Commodore 64 will be judged by.

I know you will enjoy not only the games themselves, but also the insight you will gain about programming the Commodore 64.

Melbourne House are not just publishers — we are dedicated to microcomputer software, and we are always interested in your feedback: If you have an article or program that you think might be of value to other users we want to hear from you.

Until I hear from you, let me just wish you happy programming!



ALFRED MILGROM
PUBLISHER

Contents

SPECIAL CHARACTERS

In the program listings throughout this book spaces have been used ONLY as an aid to readability. When you type the programs in on your computer these spaces should be left out to conserve memory space.

As an example the line

100 POKE 54273,I : POKE V + 23,0

would be typed as

100POKE54273,I:POKEV+23,0

on your computer. As you can see the first verson is much more legible.

The only situation in which a space should be entered into a program listing is when the symbol

" ▲ "

appears. This special character does not appear on your computer's keyboard but has been placed in the listings to tell you, the programmer, when a space is necessary.

One other symbol appears in the program listings which may not at first be recognised. This is the symbol

" : "

and is actually just the COLON symbol located on the key next to the L key.

The letter "I" and the number "1" can cause some confusion at times as well as the letter "O" and the number "0".

Once you are familiar with them they don't cause any problems, it's just a matter of recognising the difference.

The letter I looks like this: I
The number 1 looks like this: 1
The letter O looks like this: O
The number 0 looks like this: 0

CHEXSUM

RUN THIS PROGRAM FIRST

The unique CHEXSUM program validation

WHY:

When a book of programs, such as this book, is keyed in everybody invariably makes reading and typing mistakes and then spends ages trying to sort out where and what is causing the error (or errors).

Even experienced programmers often cannot identify an error by just listing the relevant line and need to do the tedious job of going back to the book, especially with DATA statements.

Realising that this is a major cause of frustration in keying the programs we decided to do something about it.

There are two short routines in this book which you should key in and save BEFORE you key any other programs in.

Using these routines you will be able to find out if you have made any keying errors at all and in which lines, before you even run the program.

In effect this means that with this book you need not waste time looking for keying errors, you simply run the routines and look at the display to identify lines containing errors. It's just that easy.

The principle behind the routines is a unique chexsum which is calculated on each individual line of the program as you have keyed it in. Compare this chexsum value with the value for that line in the list at the end of the program listing, if they are the same the line is correct, if not there is an error in that line.

HOW:

The CHEXSUM program uses line 62000 onwards, and no other program in this book uses line numbers that large.

To use it, you need to type in the CHEXSUM program first (or load it from tape). You then type in the program you have chosen from the book.

At any stage of typing your game program, you can check on the correctness of your work by entering the command RUN 62000 .

When you have completed entering your program, and are satisfied that all is correct, you can delete the CHEXSUM program by running it and answering DELETE to the question "TO PRINTER (Y, N OR DELETE) ?". You can now save your program to disk or cassette as it will still be intact in memory.

WHEN:

The simplest method is to enter the CHEXSUM program in now and save it to tape or disk. Each time you want to enter a new program just load CHEXSUM and key the program.

You can type in the CHEXSUM program at any time, even if you have started to type in a program. You cannot, of course, load in CHEXSUM from tape or disk because it will erase all you have typed so far.

A partial solution to this is the MERGE program included here that will allow you to merge two programs. To use it, you will have to save all you have typed in so far onto your cassette or disk, load the MERGE program, and run it. You will be prompted to load in your program and the CHEXSUM program.

If you are using a Datasette it might be an idea to save CHEXSUM and MERGE on a seperate cassette, saving CHEXSUM first. This will enable you to save all your corrected programs on one cassette and allow you easy access to the CHEXSUM and MERGE routines without having to constantly run backwards and forwards through the tape.

HOW CAN YOU TELL IF CHEXSUM HAS BEEN CORRECTLY ENTERED?

After having keyed CHEXSUM (or MERGE) the logical thing would be to chexsum these programs too, to ensure that they are correct. But is it possible to do this? If you follow the instructions you will be able to check them both.

Type and save CHEXSUM

Edit line 62020 and change the value of E from 61999 to 62200

Type RUN and check output against the table of values at the end of the program.

If the program is incorrect then edit the incorrect lines and resave the program after having reset the values of E in line 62020.

Type and save MERGE.

You can only check the data in the MERGE routine, however this is the most important part of the routine. All you need to do is "RUN" merge, it will check its own data and prompt you if there is an error.

Here are the routines and instructions on how to use them:

CHEXSUM ROUTINE

```
62000 T=PEEK(62)*256+PEEK(61)+1
62010 INPUT"TO PRINTER ( Y , N OR DELETE ) ";Q$:
      IF Q$="DELETE"THEN62200
62011 IF Q$<>"Y"THEN 62020
62015 CLOSE4,4:OPEN4,4:CMD4:PRINT CHR$(1);CHR$(129)
62020 PRINT CHR$(147);"CHECK SUM: -" :
      LINK=PEEK(44)*256+PEEK(43):E=61999
62100 REM * MAIN LOOP *
62120 T=LINK
62130 LINK=PEEK(T+1)*256+PEEK(T)
62135 LN=PEEK(T+3)*256+PEEK(T+2)
62136 IF LN>E THEN PRINT:PRINT"TOTAL=";CH:CLOSE4,4:END
62137 S$=STR$(LN):L=LEN(S$)-1:S$=MID$(S$,2,L)
62138 PRINT SPC(6-L);S$;
62140 CS=0 : N=0 : C=0
62150 FOR P=T+4 TO LINK-2 :PK=PEEK(P)
62160 IF PK=143 THEN P=LINK-2:GOTO 62190
62165 IF PK=34 THEN C=(C=0)
62170 IF C=0 AND PK=32 THEN 62190
62180 IF PK=137 THEN N=N+1:CS=CS+(203ORN):PK=164
62185 N=N+1:CS=CS+(PK OR N)
62190 NEXT P :CH=CH+CS : PRINT
      "=";RIGHT$(STR$(CS),LEN(STR$(CS))-1):GOTO 62120
62200 POKET,0:POKET+1,0:POKE46,PEEK(46)-2:CLR
62999 REM
```

INSTRUCTIONS

1: RUN 62000
2: answer the question.
   "Y" if you want listing on printer
   "N" if listing is to be displayed
   "DELETE" to delete CHEXSUM
3: hit "RUN/STOP" to halt output.
4: type "CONT" to continue.
5: if all is correct then "RUN 62000" again and answer with "DELETE".
6: otherwise correct errors and start again from 1.

CHEXSUM

```
62000=1348          62170=1554
62010=4494          62180=3320
62011=1281          62185=1362
62015=2125          62190=4228
62020=4255          62200=2275
62100=0
62120=577           TOTAL= 45567
62130=2067
62135=2115
62136=2530
62137=2908
62138=931
62140=1142
62150=2251
62160=2373
62165=1431
```

MERGE ROUTINE

```
10 POKE 55,0:POKE56,159:CLR
20 S=40705:FOR J=S TO S+78:READ V
30 C=C+V:POKE J,V:NEXT
40 IF C<>8756 THEN PRINT "DATA▲ ERROR":END
120 DATA 169,0,133,10,32,212,225,165
130 DATA 43,72,165,44,72,56,165,45
140 DATA 233,2,133,43,165,46,233,0
150 DATA 133,44,169,0,133,185,166,43
160 DATA 164,44,169,0,32,213,255,176
170 DATA 14,134,45,132,46,32,51,165
180 DATA 104,133,44,104,133,43,96,170
190 DATA 201,4,144,244,240,10,104,133
200 DATA 44,104,133,43,24,108,0,3
210 DATA 164,186,136,240,209,208,239
300 NEW
```

INSTRUCTIONS

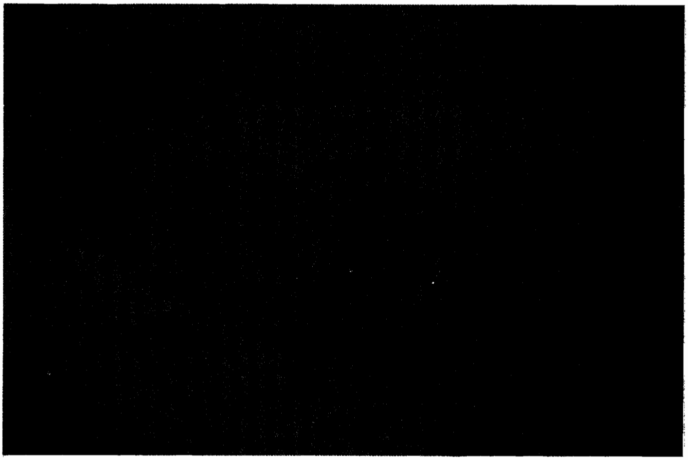
1: LOAD and RUN MERGE.
2: type or LOAD your program.
3: type SYS 40705 "CHEXSUM". (SYS 40705,8 for Disk.)
4: then follow CHEXSUM instructions.

Because MERGE, on running it, stores itself above BASIC memory it will reside there until you switch off the machine or change the top of BASIC RAM. From this it follows that you don't need to continually reload MERGE after running it the first time.

CHEXSUM AND GRAPHICS CHARACTERS

At times it might be difficult to recognise the difference between two graphics characters and when you run the CHEXSUM you will get a variance in the values. If, after having checked the line carefully, you come to the conclusion that the error must be in one of the graphics characters and you are reasonably sure that you keyed what you think the character should be, then don't worry about the CHEXSUM variance. When you run the program it will be obvious from the screen display what the character should be if it is an important character. Some characters don't really matter that much and at times you probably won't even be able to find the wrong one in the screen display.

Paranoid



Just sit back and watch the machine print a kaleidoscope of graphics patterns. When you've had enough just 'stop' the program. For an entirely different pattern re-run the program.

This is an easy, short program that has pretty impressive results. Use it in understanding your machine's graphics capabilities or just something to occupy your screen (and spectators) while you are doing something else.

PROGRAM STRUCTURE	LINES
INITIALIZATION	1 — 100
PLOT POINTS REVOLVING AROUND X, Y	105, 107
CHANGE X, Y AND CHECK IF IN SCREENS SIZE	110, 115
RANDOMLY CALCULATE DX, DY	120, 140
LOOP	145
SUBROUTINE PLOT X1, Y1	1000, 1010

VARIABLES
V = VIDEO CHIP
X, Y = BASIC SCREEN COORDINATES
DX, DY = OFFSETS FROM PREVIOUS X, Y
X1, Y1, YA, YB, XA, XB = TEMPORARY STORAGE
P(7) = BIT POSITIONS
XC = BIT TO BE SET
P = LOCATION OF BYTE TO BE PLOTTED TO

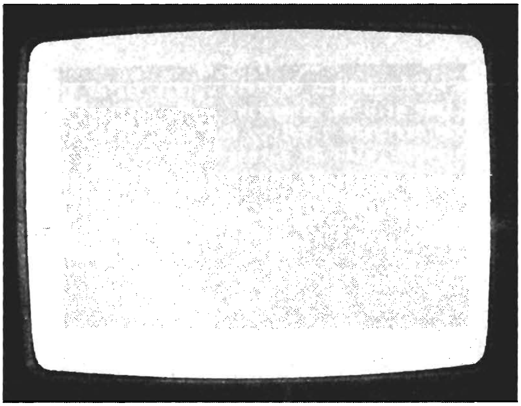
PARANOID

```
   0 BACKGROUND = 1
   5 POKE 55,255 : POKE 56,31
   6 DIM P ( 7 ) : FOR I = 0 TO 7 :
      P ( I ) = 2 ↑ ( 7 - I ) : NEXT
  10 V = 53248 : POKE V + 32,0 : POKE V + 33,0
  30 POKE V + 24, PEEK ( V + 24 ) OR 8
  40 POKE V + 17, PEEK ( V + 17 ) OR 32
  50 FOR I = 1024 TO 2024 : POKE I,BA : NEXT
  60 FOR I = 8192 TO 8192 + 8 * 1024 : POKE I,0 :
      NEXT
 100 X = 79 : Y = 49 : DX = INT ( RND ( 1 ) * 3 - 1 )
      : DY = INT ( RND ( 1 ) * 3 - 1 ) :
      IF DX = 0 AND DY = 0 THEN 100
 105 Y1 = Y : X1 = X : GOSUB 1000 : X1 = 319 - X :
      GOSUB 1000 : Y1 = 199 - Y : GOSUB 1000 :
      X1 = X : GOSUB 1000
 107 Y1 = Y * 2 : X1 = X * 2 : GOSUB 1000 :
      Y1 = 199 - Y1 : X1 = 319 - X1 : GOSUB 1000
 110 X = X + DX : Y = Y + DY : IF X < 0 OR X > 159
      THEN DX = - DX : GOTO 110
 115 IF Y < 0 OR Y > 99 THEN DY = - DY : GOTO 110
 120 IF RND ( 1 ) > .9 THEN
      DX = INT ( RND ( 1 ) * 3 - 1 )
 130 IF RND ( 1 ) > .9 THEN
      DY = INT ( RND ( 1 ) * 3 - 1 )
 135 IF DX < > 0 OR DY < > 0 THEN 105
 140 DX = INT ( RND ( 1 ) * 3 - 1 ) :
      DY = INT ( RND ( 1 ) * 3 - 1 ) :
      IF DX = 0 AND DY = 0 THEN 140
 145 GOTO 105
1000 YA = INT ( Y1 / 8 ) : YB = Y1 - YA * 8 :
      XA = INT ( X1 / 8 ) : XB = X1 - XA * 8
1005 P = 8 * 1024 + YA * 320 + XA * 8 + YB :
      XC = P ( XB )
1010 POKE P, PEEK ( P ) OR XC : RETURN
```

CHEXSUM

0=1014	110=4168
5=883	115=2536
6=2548	120=2433
10=1973	130=2433
30=1482	135=1854
40=1536	140=4665
50=1742	145=525
60=2278	1000=4565
100=5606	1005=3199
105=4985	1010=1205
107=4044	
	TOTAL= 55784

Poetry



Sit back and watch as the program writes poetry before your very eyes. Try making changes to the poem's words to make it ridiculous or humourous or just try and give it more to write.

Press'N' key to stop the program or any other key for another poem.

PROGRAM STRUCTURE	LINES
INITIALIZE ARRAYS	20 — 30
INITIALIZE SCREEN	50
CHOOSE WORDS	55 — 70
CHOOSE LINES	75
PRINT LINES OF POETRY	80 — 130
GO AGAIN?	500 — 515
WORDS DATA	1000 — 1045

VARIABLES
AJ$ = ARRAY CONTAINING ADJECTIVES FOR LINE
AV$ = SAME AS AJ$ EXCEPT ADVERBS FOR LINE
FL = TO REPEAT A LINE
MO = RAMDOM TO SET "MOOD"
AJ, AV = RANDOM TO CHOOSE WORDS

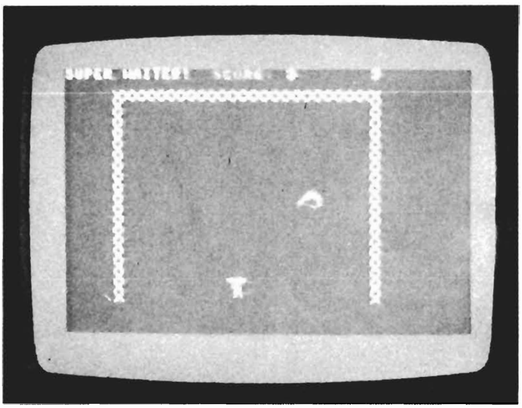
```
 10 REM * * POETRY * *
 20 DIM AJ$ ( 9,2 ) ,AV$ ( 9,2 ) : FOR I = 0 TO 9
 30 READ AJ$ ( I,0 ) ,AJ$ ( I,1 ) ,AJ$ ( I,2 )
      ,AV$ ( I,0 ) ,AV$ ( I,1 ) ,AV$ ( I,2 ) : NEXT
 50 PRINT "▣"; SPC( 14 ) ;"**▲POETRY▲**▣";
      CHR$ ( 8 )
 55 FL = 0 : MO = INT ( RND ( 1 ) * 3 )
 60 FOR I = 1 TO 4 + INT ( RND ( 1 ) * 4 ) :
      AV = INT ( RND ( 1 ) * 10 )
 65 AJ = INT ( RND ( 1 ) * 10 ) :
      IF AJ = AV THEN 65
 70 AJ$ = AJ$ ( AJ,MO ) : AV$ = AV$ ( AV,MO )
 75 ON INT ( RND ( 1 ) * 11 + 1 )
      GOTO 80,85,90,95,100,105,110,115,120,125,130
 80 PRINT "THE▲";AJ$;"▲MAN▲";AV$;
      "▲BEGUILED▲HER" : GOTO 500
 85 PRINT "A▲";AJ$;"▲WOMAN▲ENCHANTED▲ME▲WITH
      ▲";AV$;"▲BLINKING▲EYES" : GOTO 500
 90 PRINT "IN▲";AV$;"▲KEEPING▲WITH▲HER▲";AJ$;
      "▲VOWS" : GOTO 500
 95 PRINT "ALAS,▲SHE▲MUST▲";AV$;"▲LEAVE▲HIS
      ▲";AJ$;"▲PRESENCE" : GOTO 500
100 PRINT "A▲BREATH▲OF▲";AJ$;"▲AIR▲";AV$;"
      ▲RUSTLED▲IN▲THE▲TREES" : GOTO 500
105 PRINT "ANOTHER▲";AJ$;"▲DAY▲";AV$;"▲ENDED" :
      GOTO 500
110 PRINT "THE▲";AJ$;"▲HILLS▲MARCHED▲";AV$;
      "▲ACROSS▲THE▲HORIZON" : GOTO 500
115 PRINT "AND▲THEN:" : GOTO 500
120 PRINT "THE▲";AJ$;"▲BELL▲TOLLED▲";AV$;
      "▲ONCE▲AGAIN" : GOTO 500
125 PRINT "THE▲";AV$;"▲";AJ$;"▲HUMAN▲ARRIVED" :
      GOTO 500
130 PRINT "LIFE▲";AV$;"▲DAWNED▲ON▲THE▲";AJ$;
      "▲UNIVERSE" : GOTO 500
500 NEXT I : IF FL = 0 AND RND ( 1 ) > .5 THEN
      FL = 1 : GOTO 60
505 GET A$ : IF A$ = "N" THEN PRINT CHR$ ( 9 ) :
      END
510 IF A$ = "" THEN 505
515 GOTO 50
1000 DATA SORROWFUL,APPATHETIC,JOYFUL,SADLY
      ,CARELESSLY,HAPPILLY
1005 DATA PUTRID,ODOURLESS,SCENTED,FOULLY
      ,CAREFULLY,SWEETLY
1010 DATA BORING,ENLIGHTENING,INTERESTING,TIRELESSLY
      ,EFFORTLESSLY,EASILY
1015 DATA UGLY,PLAIN,BEAUTIFUL,CLUMSILY,,GRACEFULLY
1020 DATA FAT,THIN,LEAN,NOISILY,QUIETLY,
1025 DATA PATHETIC,ORDINARY,SUPER,PATHETICALLY
      ,, SUPERBLY
1030 DATA IRRITATING,CALMING,EXCITING,SHARPLY
      ,COOLY,EXCITEDLY
1035 DATA TORTUOUS,HATING,LOVING,,CONTEMPTUOUSLY
```

```
        ,LOVINGLY
1040 DATA DYING,LIVING,NEWLY BORN,PAINFULLY,EASILY,
1045 DATA STUPID,IGNORANT,INTELLIGENT,FOOLISHLY
        ,,INTELLIGENTLY
```

CHEXSUM

```
     10=0
     20=1889
     30=3890
     50=2263
     55=1640
     60=3083
     65=2234
     70=2233
     75=3848
     80=3558
     85=5647
     90=3916
     95=4869
    100=5163
    105=3316
    110=5313
    115=1455
    120=4107
    125=3351
    130=4283
    500=2791
    505=1837
    510=852
    515=476
   1000=5167
   1005=4661
   1010=6211
   1015=3690
   1020=2536
   1025=4266
   1030=4885
   1035=4533
   1040=3581
   1045=5021
TOTAL= 116565
```

Turkey



Attempt to keep the turkey in the air as long as possible by manoeuvering the waiter underneath it and then bouncing it off the platter.

You control the waiter with the 'Z' (left) and 'M' (right) keys. The turkey can be dropped twice but becomes inedible after the third time and the game ends.

PROGRAM STRUCTURE	LINES
INITIALIZATION	1 — 5
POINT SCORE AND NO. OF DROPS LEFT	10
READ KEY	20
MOVE LEFT	26
MOVE RIGHT	32
MOVE TURKEY	50
CHECK IF WAITER CAUGHT TURKEY	95 — 100
ANOTHER GAME	120
IF YES, RUN AGAIN	140
ELSE END	160
DATA	9000 — 9050

VARIABLES
V = VIDEO CHIP
CO = DIFFERENCE BETWEEN VIDEO DISPLAY AND SCREEN MEMORY
LI = NO. OF DROPS LEFT
P = POSITION OF WAITER
X, Y = POSITION OF TURKEY
S = SCORE
PE = KEY PRESSED
XD, YD = DISPLACEMENTS FOR TURKEY
M, X1, Y1, P = TEMPORARY STORAGE

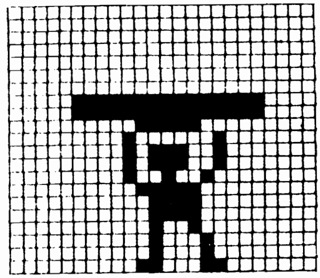
TURKEY

```
   1 POKE 53281,0 : POKE 53280,9
   2 CO = 54272 : P = 150 : X = 150 : Y = 190 :
       V = 53248 : SC = 0 : LI = 3 :
       XD = INT ( RND ( 1 ) * 9 - 4 ) : YD = - 1
   3 POKE V + 21,3 : POKE 2040,13 : POKE 2041,14
   4 FOR I = 0 TO 126 : READ Q : POKE 832 + I,Q :
       NEXT
   5 POKE V,P : POKE V + 1,200 : POKE V + 2,X :
       POKE V + 3,Y
   7 PRINT " ■ ■SUPER▲WAITER!▲ ▲ ■SCORE:"
   8 FOR I = 55381 TO 55406 : POKE I,5 :
       POKE I - CO,86 : NEXT :
       FOR I = 55381 TO 56156 STEP 40
   9 POKE I,5 : POKE I - CO,86 : POKE I + 25,5 :
       POKE I + 25 - CO,86 : NEXT
  10 PRINT "■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■
       ■ ■ ■ ■ ■ ■ ■ ■ ■ ■";S;" ▲ ▲ ▲ ▲ ▲";LI
  20 PE = PEEK ( 197 )
  22 IF PE < > 12 AND PE < > 36 THEN 50
  26 IF PE = 12 AND P > 70 THEN P = P - 2
  32 IF PE = 36 AND P < 240 THEN P = P + 2
  40 POKE V,P
  50 Y1 = Y : X1 = X : M = 0
  60 Y1 = Y1 + YD : X1 = X1 + XD
  70 IF X1 < 70 OR X1 > 240 THEN XD = - XD :
       S = S + 1 : M = 1
  80 IF Y1 > 220 THEN X = P : Y = 200 :
       XD = INT ( RND ( 1 ) * 9 - 4 ) : YD = - 1 :
       LI = LI - 1 : GOTO 110
  90 IF Y1 < 70 THEN YD = 1 :
       XD = INT ( RND ( 1 ) * 9 - 4 ) : S = S + 1 :
       M = 1
  95 IF Y1 < 210 AND PEEK ( V + 30 ) < > 0 THEN
       YD = - 1 : XD = INT ( RND ( 1 ) * 9 - 4 ) :
       S = S + 2
 100 IF M = 0 THEN X = X1 : Y = Y1 : POKE V + 2,X :
       POKE V + 3,Y
 110 IF LI > 0 THEN GOTO 10
 120 PRINT "■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■THE
        ▲TURKEY▲IS▲INEDIBLE!!"
 130 PRINT "■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ▲ ▲ ▲WANT▲TO
        ▲TRY▲AGAIN?"
 140 GET A$ : IF A$ = "Y" THEN RUN
 150 IF A$ < > "N" THEN 140
 160 END
9000 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
       ,0,0,0,0,0,7,255,240,7,255,240,0,62,0,0
9010 DATA 64,128,0,92,128,0,92,128,0,73,0
       ,0,62,0,0,30,0,0,30,0,0,18,0,0,17,0,0,17
9020 DATA 0,0,49,128
9030 DATA 0,0,,0,0,0,0,0,0,0,0,0,192,0,7
       ,240,0,15,240,0,31,252,0,63,254,0,127,254
9040 DATA 0,255,127,1,248,63,9,224,14,7,128
       ,14,15,240,15,3,14,30,6,1,248,0,0,112
9050 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0
```

CHEXSUM

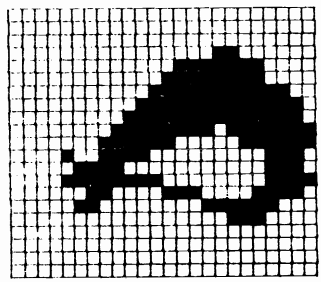
```
   1=1179
   2=6280
   3=1833
   4=2100
   5=2520
   7=2064
   8=4206
   9=3215
  10=1720
  20=796
  22=1887
  26=1942
  32=1991
  40=368
  50=1287
  60=1712
  70=3238
  80=5506
  90=3815
  95=4934
 100=2333
 110=1187
 120=2713
 130=2288
 140=1263
 150=1125
 160=129
9000=4460
9010=4624
9020=660
9030=4589
9040=4559
9050=1473

TOTAL= 84596
```



SPRITE

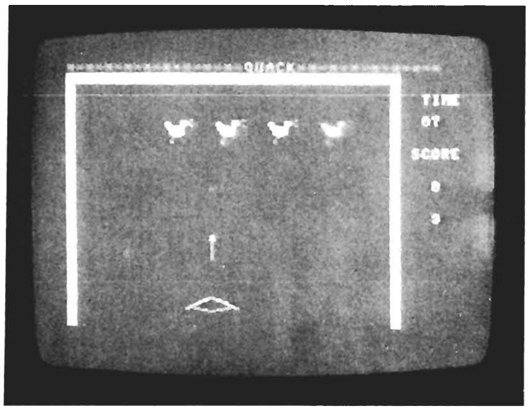
WAITER



SPRITE

TURKEY

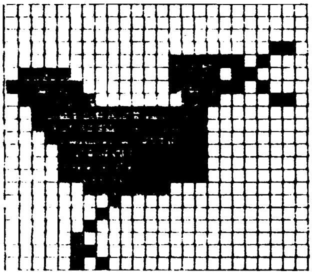
Quack



This is the old fair ground game but instead of using bullets you are equipped with a bow and arrow. An arrow will of course travel slower than a bullet and thus requires better anticipation. The ducks do not move at a uniform speed which makes the game a lot harder. You have a limited number of arrows and also a time limit.

Your controls are:

- 'Z' left
- 'C' right
- 'M' to fire an arrow

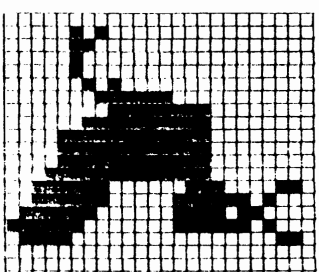


SPRITE

DUCK

PROGRAM STRUCTURE	LINES
INITIALIZATION	0 — 9
GET KEY PRESSED	10
PRINT TIME, SCORE, ETC	11 — 12
IF OUT OF TIME GO TO END	
MOVE BOW AND ARROW	
IF 'S' FIRE ARROW	15
IF SPRITE COLLISION FLIP BIRD	18
POSITION BOW AND ARROW	21
MOVE DUCKS	30 — 40
FLIP BIRD ROUTINE	100 — 130
PRINT MESSAGES	200 — 280
FIRE ARROW ROUTINE	300 — 310
DATE	9000 — 9110

VARIABLES
S = SCORE
A1 = ARROWS LEFT
AR = 0: NOT IN FLIGHT
P = BOW POSITION
X, Y = ARROW POSITION
X ARRAY = DUCK POSITIONS
PE = CURRENT KEY
Z = SPRITE COLLISION



SPRITE

DEAD
DUCK

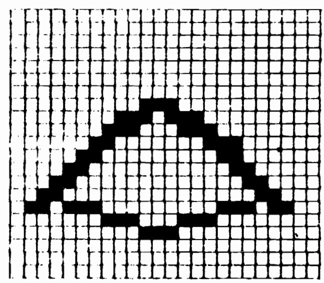
QUACK

```
  0 POKE 55,255 : POKE 56,47 : CO = 54272 :
     A1 = 10 : S = 0 : POKE 54296,15 :
     POKE 54278,240 : POKE 54276,3
  1 PRINT "{CLR}" : POKE 53281,0 : POKE 53280,0 :
     V = 53248 : POKE V + 21,63 : POKE 2040,192 :
     POKE 2041,193
  2 FOR I = 1 TO 4 : POKE 2041 + I,194 :
     POKE V + 40 + I,2 + I : NEXT :
     POKE V + 39,8 : POKE V + 40,1 : POKE V + 29,1
  3 FOR I = 0 TO 254 : READ Q : POKE 12288 + I,Q :
     NEXT
  4 P = 120 : X = 132 : Y = 217 : POKE V,120 :
     POKE V + 1,220 : POKE V + 2,X :
     POKE V + 3,Y : FOR I = 0 TO 3
  5 X ( I + 1 ) = ( I + 1 ) * 40 : Y ( I + 1 ) = 90
     : POKE V + 4 + I * 2,X ( I + 1 ) :
     POKE V + 5 + I * 2,Y ( I + 1 ) : NEXT
  6 I = PEEK ( V + 30 ) : FOR I = 55376 TO 56256
     STEP 40 : POKE I,4 : POKE I - CO,160 :
     POKE I + 32,4
  7 POKE I - CO + 32,160 : NEXT :
     FOR I = 55336 TO 55368 : POKE I,4 :
     POKE I - CO,160 : NEXT
  8 TI$ = "000000" : PRINT "{HOME}{RED}***********
     *********{GREEN}QUACK{RED}*****************"
  9 PRINT "{HOME}{DOWN}{DOWN}{DOWN}{DOWN}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{WHITE}
     TIME{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}SCORE"
 10 PE = PEEK ( 197 )
 11 PRINT "{HOME}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{PI}";
     RIGHT$ ( TI$,2 ) : PRINT "{HOME}{DOWN}{DOWN}
     {DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}";S;
 12 PRINT "{DOWN}{DOWN}{DOWN}{RIGHT}{RIGHT}▲ ▲{RIGHT}{RIGHT}{RIGHT}";A1 :
     IF TI$ = "000100" THEN 200
 13 IF PE = 12 AND P > 35 THEN P = P - 4 :
     IF AR = 0 THEN X = X - 4
 14 IF PE = 20 AND P < 237 THEN P = P + 4 :
     IF AR = 0 THEN X = X + 4
 15 IF PE = 36 AND AR = 0 AND A1 > 0 THEN
     A1 = A1 - 1 : AR = 5 : GOSUB 300
 18 Y = Y - AR : Z = PEEK ( V + 30 ) :
     IF Z > 3 AND Y > 90 THEN GOSUB 100
 19 IF Y < 60 THEN Y = 217 : AR = 0 : X = P + 12 :
     POKE V + 3,Y
 20 POKE V,P : POKE V + 2,X : IF AR > 0 THEN
     POKE V + 3,Y
 30 FOR I = 0 TO 3 : X ( I + 1 ) = X ( I + 1 ) + 3 :
     IF X ( I + 1 ) > 241 THEN X ( I + 1 ) = 35 :
     POKE 2042 + I,194
 40 POKE V + 4 + I * 2,X ( I + 1 ) :
     POKE V + 5 + I * 2,Y ( I + 1 ) : NEXT
 50 GOTO 10
100 S = S + 10 : Z = PEEK ( V + 30 ) : A = 5 :
     IF Z AND 8 THEN A = 1
105 IF Z AND 16 THEN A = 2
```

```
 110 IF Z AND 32 THEN A = 3
 113 IF Z AND 4 THEN A = 4
 115 POKE 2042 + A,195 : AR = 0 : Y = 218 :
      X = P + 12 : POKE V + 3,Y :
 120 POKE 54276,0 : POKE 54276,33
 125 FOR I = 15 TO 1 STEP - .5 : POKE 54273,I * 2 :
      POKE 54296,I : NEXT : POKE 54296,15 :
      POKE 54273,0
 130 Z = PEEK ( V + 30 ) : RETURN
 200 POKE 54276,0 : POKE 54276,17 :
      FOR I = 240 TO 1 STEP - 3 : POKE 54273,I :
      NEXT : POKE 54273,0
 210 FOR I = 1 TO 11 : POKE V + I,0 : NEXT
 220 PRINT "[CLR] [RED]YOUR TIME IS UP!"
 230 PRINT "[DOWN][DOWN][WHT]YOU SHOT";S / 10;"BIRDS."
 240 IF S > 50 THEN PRINT "[DOWN]WELL DONE!!!"
 250 PRINT "[DOWN][DOWN][DOWN][CYN]DO YOU WANT ANOTHER GO?"
 260 GET A$ : IF A$ = "Y" THEN RUN
 270 IF A$ < > "N" THEN 260
 280 END
 300 POKE 54276,0 : POKE 54276,129 :
      FOR I = 1 TO 80 STEP 2 : POKE 54273,I : NEXT
      : POKE 54273,0
 310 RETURN
9000 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
      ,0,0,0,0,0,0,56,0,0,238,0,1,199,0
9010 DATA 3,1,192,6,0,192,12,0,96,24,0,48,48
      ,0,24,110,0,236,1,199,0,0,56,0
9020 DATA 0,0,0,0,0,0,0,0,0,99
9030 DATA 0,0,0,0,0,0,0,16,0,0,56,0,0,56,0
      ,0,124,0,0,16,0,0,16,0,0,16,0,0,16
9040 DATA 0,0,16,0,0,16,0,0,16,0,0,16,0,0,16
      ,0,0,16,0,0,16,0,0,16,0,0,16,0,0,16
9050 DATA 0,0,16,0,99
9060 DATA 0,0,0,0,0,0,0,0,0,0,0,6,0,7,232
      ,120,7,176,252,7,232,126,3,134,63,255,0
9070 DATA 63,255,0,31,255,0,15,255,0,15
      ,255,0,15,255,0,3,248,0,0,128,0,1,0,0
9080 DATA 2,0,0,4,0,0,6,0,0,5,0,0,99
9090 DATA 0,0,0,5,0,0,6,0,0,4,0,0,4,0,0,2
      ,128,0,1,248,0,0,255,0,3,255,0,15
9100 DATA 255,0,15,255,0,15,255,0,31,255,0,
      63,3,134,63,7,232,126,7,176,252,7,232
9110 DATA 120,0,6,0,0,0,0,0,0,0,0,0,0,0
```

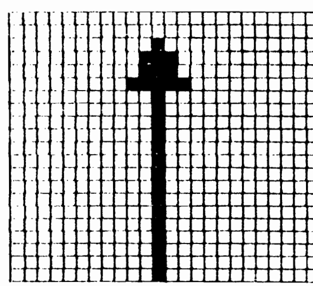
CHEXSUM

0=4717	9000=3768
1=4422	9010=3835
2=5406	9020=1190
3=2219	9030=3988
4=5124	9040=4336
5=6004	9050=702
6=4577	9060=4505
7=4137	9070=4057
8=3650	9080=1538
9=2857	9090=3794
10=796	9100=4536
11=4219	9110=1841
12=2797	
13=3375	TOTAL= 153575
14=3435	
15=3471	
18=3386	
19=3047	
20=2458	
30=5333	
40=3399	
50=472	
100=3134	
105=995	
110=997	
113=937	
115=3332	
120=1219	
125=4310	
130=1122	
200=4126	
210=1586	
220=1612	
230=2083	
240=1828	
250=2318	
260=1263	
270=1124	
280=129	
300=3926	
310=143	



SPRITE

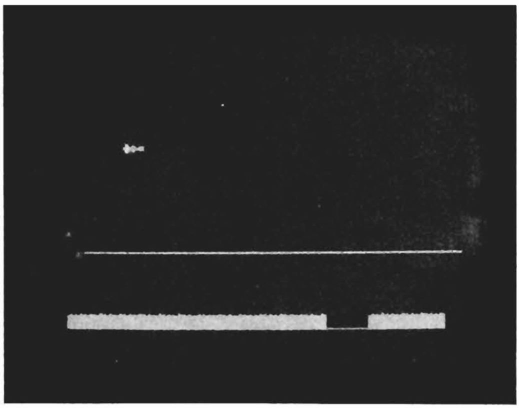
BOW



SPRITE

ARROW

Laser Tower



Escape from the Laser Tower, battling against gravity and evading the tractor beam. The tractor beam fires from the tower on the left of the screen. It alters its height as you move away from it, trying to land in the pit at the right.

Your controls are: cursor up for up, cursor right for right.

PROGRAM STRUCTURE	LINES
INITIALIZATION	5 — 35
PRINT GROUND	50 — 54
MOVE TOWER	55 — 80
FIRE LASER	85 — 95
GET MOVE	100
IF MOVEMENT MAKE NOISE	101 — 102
CALCULATE NEW POSITION	105 — 130
IF HOME: END	130, 2000 — 2025
DON'T LET GO OFF SCREEN	135 — 145
IF COLLISION THEN: END	150 — 1010
DATA	10000 — 10070

VARIABLES
GR = GRAVITY
VV = VERTICAL VELOCITY
HV = HORIZONTAL VELOCTY
FR = TRACTOR BEAM
A = X POSITION
TW = TOWER HEIGHT
PW = LASER

LASER TOWER

```
  5 POKE 55,255 : POKE 56,47
 10 V = 53248 : CO = 54272 : POKE 54296,15
 12 FOR I = 12 * 1024 + 256 TO 12 * 1024 + 263 :
      POKE I,0 : NEXT
 15 FOR I = 0 TO 47 : READ A : POKE 12 * 1024 + I,A
      : NEXT : POKE V + 22, PEEK ( V + 22 ) OR 16
 20 POKE V + 24,28 : POKE V + 34,2 : POKE V + 35,7 :
      POKE V + 33,0 : POKE V + 32,0
 25 FOR I = 12352 TO 12478 : READ A : POKE I,A :
      NEXT : POKE V + 37,2 : POKE V + 38,7
 30 POKE V,75 : POKE V + 16,0 : POKE V + 1,100 :
      POKE V + 28,0 : POKE V + 39,4 : POKE V + 21,1
 35 GR = .4 : VV = 0 : HV = 0 : FR = .5 :
      POKE 2040,193
 50 PRINT "♥ ] Q Q Q Q Q Q Q Q Q Q Q Q Q Q Q
       £A Q ] ] ] ] ] ]"; : FOR I = 1 TO 7 :
      PRINT "£ Q ▮▮"; : NEXT : TW = 1625
 51 PRINT "↑CCCCCCCCCCCCCCCCCCCCCCCC
       ▲ ▲ ▲ ▲ CCCCCCCC ] ]";
 52 PRINT "DDDDDDDDDDDDDDDDDDDDDDDDDDEEEEDDDDDDDDD";

 53 FOR T=1945 TO TW+40 STEP-40 :POKE T,0 :POKE T+CO,10
     :NEXT
 54 A = PEEK ( V + 31 )
 55 R = INT ( RND ( 1 ) * 3 + 1 ) : ON R
      GOTO 85,60,70
 60 IF TW < = 1065 THEN 100
 65 POKE TW + CO,10 : POKE TW,0 : TW = TW - 40 :
      GOTO 80
 70 IF TW > = 1945 THEN 100
 75 POKE TW,32 : TW = TW + 40
 80 POKE TW + CO,2 : POKE TW,1 : GOTO 100
 85 POKE 54276,0 : IF RND ( 1 ) < .95 THEN 100
 90 J = TW + 1 : EN = TW + 38 : POKE 54273,72 :
      POKE 54272,169 : POKE 54277,44 : POKE 54275,7
      : PW = 0
 91 POKE 54274,0 : POKE 54276,65
 93 IF PEEK ( V + 31 ) THEN 1000
 94 POKE J + CO,7 : POKE J,2 : J = J + 1 :
      POKE 54272,PW : PW = PW + 6 :
      IF J < = EN THEN 93
 95 FOR EN = TW + 1 TO J : POKE 54274,PW :
      PW = PW - 6 : POKE EN,32 : NEXT
100 K = PEEK ( 197 ) : HV = HV - FR - ( K = 2 )
101 IF K < > 2 AND K < > 7 THEN 105
102 POKE 54273,2 : POKE 54272,37 : POKE 54277,47 :
      POKE 54276,0 : POKE 54276,129
105 IF HV < - 4 THEN HV = - 4
110 IF HV > 4 THEN HV = 4
115 VV = VV + GR + ( K = 7 ) : IF VV < - 4 THEN
      VV = - 4
120 IF VV > 4 THEN VV = 4
125 POKE V + 1, PEEK ( V + 1 ) + VV :
      A = PEEK ( V + 1 ) : IF A < 48 THEN
      POKE V + 1,40
130 IF A > 234 THEN 2000
135 A = PEEK ( V ) + HV : B = PEEK ( V + 16 ) AND 1
```

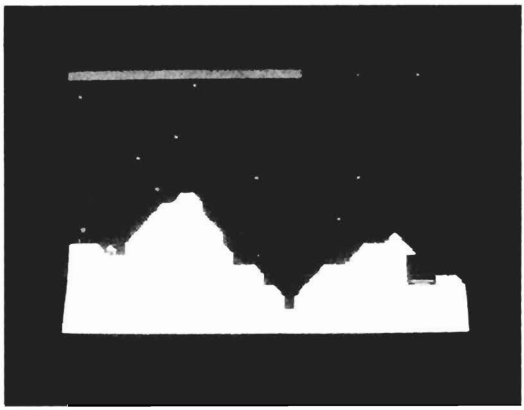
```
      : IF A > 255 THEN A = 0 : B = 1
 140 IF A < 0 AND B THEN B = 0 : A = 255
 142 IF A > 55 AND B THEN A = 55
 145 IF A < 32 AND B = 0 THEN 1000
 150 POKE V,A : POKE V + 16, PEEK ( V + 16 ) OR B
 155 IF B = 0 AND PEEK ( V + 16 ) AND 1 THEN
      POKE V + 16, PEEK ( V + 16 ) - 1
 160 IF PEEK ( V + 31 ) AND 1 THEN 1000
 165 GOTO 55
1000 POKE 2040,194 : POKE V + 28,1 : POKE 54273,34 :
      POKE 54272,75 : POKE 54277,143
1005 POKE 54276,129 : FOR I = 1 TO 500 : NEXT :
      POKE 54273,0 : POKE 54273,0 : POKE V + 24,20
1010 PRINT "{CLR}{DOWN}{DOWN}{DOWN}{DOWN}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}▓HARD▲LINES" :
      GOTO 2010
2000 FOR I = 1 TO 5000 : NEXT : POKE 54273,0 :
      POKE 54272,0 : POKE V + 24,20
2005 PRINT "{CLR}{DOWN}{DOWN}{DOWN}{DOWN}" SPC( 15 ) "WELL▲DONE!" :
      PRINT "{DOWN}{DOWN}{DOWN}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}YOU▲ESCAPED
       ▲ALIVE"
2010 PRINT "{DOWN}{DOWN}{DOWN}ANOTHER▲GAME?" : POKE V + 21,0 :
      POKE 54276,0
2015 GET A$ : IF A$ = "Y" THEN POKE V + 24,28 :
      GOTO 30
2020 IF A$ = "N" THEN END
2025 GOTO 2015
10000 DATA 220,236,220,236,220,236,220,236,0
      ,0,0,48,127,120,252,252
10005 DATA 0,0,0,0,255,0,0,0,0,0,0,8,157,255,255
      ,255,255,255,255,255,255,255
10010 DATA 255,255,0,0,0,0,0,0,0,255
10015 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
      ,0,0,0,12,32,0,14,48,128,31,121
10055 DATA 240,63,135,160,31,121,240,14,48,128
      ,12,32,0,0,0,0,0,0,0,0,0,0,0,0,0
10060 DATA 0,0,0,0,0,0,0,0,99,0,0,0,0,12,0,0,63
      ,0,0,55,0,0,55,0,252,55,0
10065 DATA 55,247,240,53,87,112,13,85,255,3
      ,85,87,0,245,95,0,213,240,0,55,0
10070 DATA 0,55,0,0,221,192,3,93,192,13,125,192
      ,13,205,192,55,3,0,0,0,0,0,0,0
```

CHEXSUM

```
5=981
10=1913
12=2958
15=4041
20=3591
25=3601
30=4135
35=2619
50=3587
51=3601
52=3720
53=3223
54=896
55=2497
60=1266
65=2794
70=1270
75=1403
80=1886
85=1772
90=4842
91=1227
93=1202
94=4580
95=3514
100=2278
101=1678
102=3285
105=1463
110=1123
115=3063
120=1143
125=3922
130=947
135=3812
140=1673
142=1295
145=1401
150=1938
155=3286
160=1444
165=477
1000=3392
1005=3806
1010=2355
2000=3038
2005=4558
2010=2643
2015=2409
2020=895
2025=580
10000=3223
10005=3967
10010=1502
10015=3988
10055=4398
10060=3490
10065=3838
10070=4070

TOTAL= 153499
```

Moon Buggy



Manoeuvre your jumping moon buggy over the moon's surface attempting to reach the landing pad on the other side without running out of fuel or crashing into a star mine.

The moon buggy needs to build up thrust but be careful because once "lift-off' has occurred it has a lot of forward momentum. The star mines are everywhere in space and you have to manoeuvre between them to get to the landing pad.

Your controls are:

'Z' left
'C' right
'M' thrust

PROGRAM STRUCTURE	**LINES**
INITIALIZATION	1 — 15
DISPLAY SET-UP	20 — 45 & 200 — 2050
GET KEYBOARD CHARACTER	50
VERTICAL CONTROL OF BUGGY	52 — 77
THRUST SOUND	56
ZERO FUEL TEST	78
HORIZONTAL CONTROL	80 — 86
MINE COLLISION TEST	87
ERASE BUGGY	90, 102
DISPLAY BUGGY	100
TEST IF BUGGY ON PAD	101
DISPLAY REMAINING FUEL	102
MINE EXPLOSION	200 — 210
NEW GAME PROMPT	220 — 235
DATA	10000 — 10025

VARIABLES
SC = SCREEN REFERENCE POINT
X1 = HORIZONTAL POSITION OF BUGGY
Y1 = VERTICAL POSITION OF BUGGY
X = PREVIOUS HORIZONTAL POSITION
Y = PREVIOUS VERTICAL POSITION
FU = FUEL
K = KEYBOARD VALUE
VV = VERTICAL VELOCITY
D = VERTICAL DISTANCE

MOON BUGGY

```
  1 POKE 55,255 : POKE 56,47 : V = 53248
      : POKE V + 32,0 : POKE V + 33,0
  5 FOR I = 12544 TO 12551 : POKE I,0 : NEXT
      : POKE 54296,15
 10 FOR I = 0 TO 111 : READ A : POKE 12288 + I,A
      : NEXT
 12 POKE V + 24,28
 15 X = 4 : Y = 17 : SC = 1024 : CO = 55296
      : KB = 197 : FU = 250 : VV = 0
 20 GOSUB 2000
 25 FOR I = 1 TO 25
 30 P = INT ( RND ( 1 ) * 630 )
      : IF PEEK ( SC + P ) < > 32 THEN 30
 35 POKE CO + P,10 : POKE SC + P,12 : NEXT I
 37 PRINT "{S}{£}CCCCCCCCCCCCCCCCCCCCCCCC";
 40 PRINT
      "{S}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{]}{_}D
       {Q}{▮}{▮}EFG";
 45 PRINT
      "{S}{Q}{▮}{▮}{▮}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}
       {Q}{Q}{Q}{▮}{▮}{▮}{↑}HIJ";
 50 K = PEEK ( KB ) : IF FU < 2 THEN 60
 52 IF K < > 36 THEN 60
 55 VV = VV - 1 : FU = FU - 2
      : IF VV < - 5 THEN VV = - 5
 56 POKE 54273,72 : POKE 54272,169 : POKE 54277,15
      : POKE 54276,0 : POKE 54276,129
 60 VV = VV + .5 : IF VV > 10 THEN VV = 10
 65 D = VV / 10 : Y1 = INT ( Y + D )
      : P = PEEK ( SC + Y1 * 40 + X )
      : IF P = 2 AND VV > 0 THEN FU = FU - .05
      : VV = 0 : D = 0
 67 IF P = 12 THEN 200
 70 IF P = 0 THEN X1 = X - 1 : Y1 = Y + 1
      : FU = FU - .1 : GOTO 86
 75 IF P = 1 THEN X1 = X + 1 : Y1 = Y + 1
      : FU = FU - .1 : GOTO 86
 77 Y1 = Y + D : IF Y1 < 1 THEN Y1 = 1 : VV = 0
 78 IF FU < 1 THEN 400
 80 X1 = X + ( K = 12 ) - ( K = 20 )
      : FU = FU + ( K = 12 ) + ( K = 20 )
      : IF X1 < 0 THEN X1 = 0
 85 IF X1 > 39 THEN X1 = 0
 86 IF X = X1 AND INT ( Y1 ) = INT ( Y ) THEN 90
 87 IF PEEK ( SC + INT ( Y1 ) * 40 + X1 ) = 12
      THEN 200
 88 IF PEEK ( SC + INT ( Y1 ) * 40 + X1 ) < > 32
      THEN 100
 90 POKE SC + INT ( Y ) * 40 + X,32 : X = X1 :Y = Y1
100 Y1 = INT ( Y ) : POKE CO + Y1 * 40 + X,7
      : POKE SC + Y1 * 40 + X,11
101 IF X > 33 AND X < 37 AND INT ( Y ) = 19
      THEN 300
102 POKE SC + FU / 10,32
```

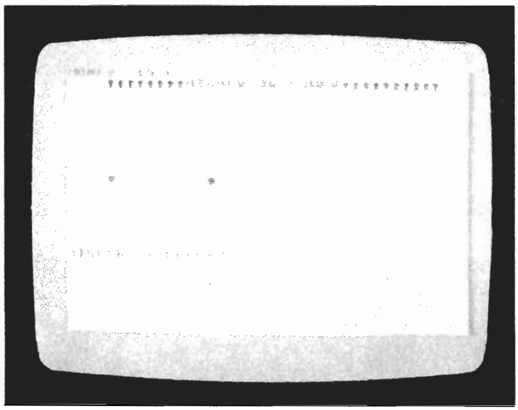
```
105 GOTO 50
200 POKE SC + INT ( Y ) * 40 + X,32
     : Y = INT ( Y1 ) : POKE CO + Y * 40 + X,2
     : POKE SC + Y * 40 + X,13
205 POKE 54273,34 : POKE 54272,75 : POKE 54277,127
     : POKE 54276,0 : POKE 54276,129
210 FOR I = 1 TO 2000 : NEXT : POKE V + 24,20
     : PRINT
       "{CLR}{?}{?}YOU▲CRASHED▲INTO▲A▲MINE{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}"
220 PRINT "{?}ANOTHER▲GAME?" : POKE 54272,0
     : POKE 54273,0
225 GET A$ : IF A$ = "Y" THEN 12
230 IF A$ < > "N" GOTO 225
235 END
300 FOR I = 1 TO 2000 : NEXT
305 POKE V + 24,20
     : PRINT
       "{CLR}{?}{?}WELL▲DONE,▲YOU▲COMPLETED▲YOUR▲
        MISSION{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}"
310 GOTO 220
400 POKE V + 24,20 : FOR I = 1 TO 1000 : NEXT
     : PRINT
       "{CLR}YOU▲RAN▲OUT▲OF▲FUEL!{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}"
     : GOTO 220
2000 PRINT
     "{CLR}{WHT}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}
      {RIGHT}{RIGHT}{RIGHT}{RIGHT}@BA{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}
      {RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}@B";
2005 PRINT
     "CCC{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}
      {RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}@CCCCCA
      {RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}";
2010 PRINT
     "{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}@CCCCCCCA{RIGHT}{RIGHT}{RIGHT}
      {RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}
      {RIGHT}BBB{RIGHT}{RIGHT}{RIGHT}@CCCCCCCCC{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}";
2015 PRINT
     "{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}@A{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}CCCA
      {RIGHT}{RIGHT}CCCCCCCCCA{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}
      {RIGHT}@BBCCA{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}";
2020 PRINT
     "CCCCBBCCCCCCCCCCC{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}
      {RIGHT}@CCCCC{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}CCCCCCCCCCCCCCCCBBA
      {RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}@";
2025 PRINT
     "BBCCCCCC{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}CCCCCCCCCCCCCCCCCCC
      {RIGHT}{RIGHT}{RIGHT}{RIGHT}@CCCCCCCCCBBBBBACCCCCCCCCCC";
2030 PRINT
     "CCCCCCCCCCA{RIGHT}@CCCCCCCCCCCCCCCCCCCCCCC
      CCCCCCCCCCCCCCCCC{RIGHT}CCCCCCCCCCCC";
2035 PRINT "CCCCC";
2040 PRINT
     "CCCCCCCCCCCCCCCCCCCCCCCBCCCCCCCCCCCCCCCCCC";
2045 PRINT
     "CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC";
     : POKE 2023,3
```

```
2050 POKE 56295,1 : RETURN
10000 DATA 1,3,7,15,15,31,95,255,128,192,224,240,248,
      248,252,255
10005 DATA 149,255,255,255,255,255,255,255,255,255,
      255,255,255,255,255,255
10010 DATA 0,0,24,126,126,60,24,36,0,0,1,2,2,2,5,5,
      66,129,0,0,0,0,0,0
10015 DATA 0,0,128,64,64,64,160,160,0,0,0,0,63,68,
      131,4,0,0,0,0,255,0,255,24
10020 DATA 0,0,0,0,252,34,193,32,0,120,68,254,255,
      66,231,66
10025 DATA 0,0,0,16,40,16,0,0,8,56,252,63,60,52,34,64
```

CHEXSUM

```
1=3071       101=2279      10025=2473
5=2467       102=1082
10=2201      105=473       TOTAL= 171732
12=672       200=5231
15=3932      205=3417
20=347       210=5069
25=710       220=2450
30=2994      225=1243
35=1815      230=1327
37=2328      235=129
40=1583      300=1020
45=2107      305=5190
50=1598      310=527
52=969       400=5072
55=3204      2000=3823
56=3417      2005=4092
60=2150      2010=4889
65=6957      2015=4241
67=843       2020=5376
70=3501      2025=6222
75=3501      2030=6515
77=2283      2035=639
78=867       2040=3701
80=4910      2045=4398
85=1136      2050=736
86=2012      10000=3074
87=2426      10005=3759
88=2612      10010=3373
90=2522      10015=3936
100=3544     10020=2787
```

Higher or Lower



Bet on the card being higher or lower than the previous one. If you win, you may get the chance to go for double or nothing.

The routines used to draw cards and select them from the deck are good. They would be useful in creating programs for your own card games.

All instructions necessary for the game are in the program.

PROGRAM STRUCTURE	**LINES**
INITIALIZE — SCREEN	0 — 1
— DECK	4 — 8
WAIT FOR DEAL COMMAND	10
PRINT MONEY	11
PICK A CARD	14 — 16
PRINT CARD	17 — 37
BUZZ	38 — 39
IF WIN	40
MUSIC AND MESSAGES	50 — 52
PLAY AGAIN?	55 — 75
ELSE MESSAGES AND PLAY AGAIN?	80 — 95
CLEAR BOTTOM OF SCREEN	100 — 110

VARIABLES

S$ = SYMBOL OF SUITS
C = PACK OF CARDS
M = MONEY
D = DOUBLE OR NOTHING
CU = HOW MANY TURNS
D = POINTER INTO PACK

HIGHER OR LOWER

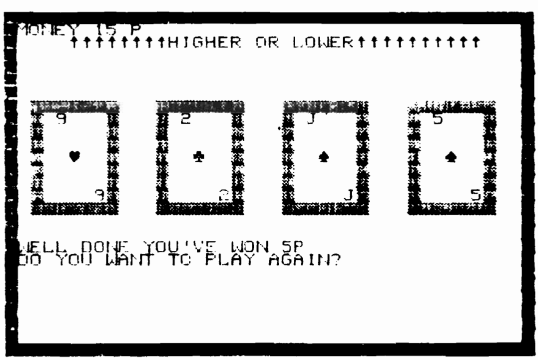
```
 0 PRINT "{CLR}" : POKE 53281,7 : POKE 53280,6
     : DIM S$ ( 52 ) : DIM C ( 52 ) : M = 20 : D = 0
     : CU = 0
 1 POKE 54296,15 : POKE 54276,0 : POKE 54278,240
 4 FOR J = 0 TO 3 : FOR I = 1 TO 13
     : C ( I + J * 13 ) = I : S$ ( I + J * 13 )
       = "{BLK}♠"
 5 IF J = 1 THEN S$ ( I + J * 13 ) = "{RED}♥"
 6 IF J = 2 THEN S$ ( I + J * 13 ) = "{BLK}♣"
 7 IF J = 3 THEN S$ ( I + J * 13 ) = "{RED}♦"
 8 NEXT I,J
10 POKE 54276,0 : POKE 54276,17
11 POKE 54273,34 : POKE 54272,75 : PRINT "{HOME}{RED}MONEY
      ▲ ▲ ▲ ▲ ▲ ▲ ▮▮ ▮▮ ▮▮ ▮▮ ▮▮ ▮▮ ";M;"P" : GOSUB 100
13 POKE 54273,0 : POKE 54272,0 : CU = CU + 1
14 FOR I = 1 TO 4
15 P = INT ( RND ( 1 ) * 52 + 1 ) : IF C ( P ) = 0
     THEN 15
16 N1 = N : N$ = STR$ ( C ( P ) ) : N = C ( P )
17 IF C ( P ) = 1 THEN N$ = " ▲A"
18 IF C ( P ) = 11 THEN N$ = " ▲J"
19 IF C ( P ) = 12 THEN N$ = " ▲Q"
20 IF C ( P ) = 13 THEN N$ = " ▲K"
21 IF C ( P ) = 10 THEN N$ = "10"
22 C ( P ) = 0 : PRINT "{HOME}{DOWN}▲ ▲ ▲ ▲ {BLK}↑↑↑↑↑↑↑↑↑{RED}
     HIGHER ▲OR ▲LOWER{BLK}↑↑↑↑↑↑↑↑↑↑↑"
23 IF I = 1 THEN 27
24 PRINT "{HOME}{17 DOWN}{LEFT}
     HIGHER ▲OR ▲LOWER?"
25 PRINT "{HOME}{17 DOWN}{LEFT}
     HIGHER ▲OR ▲LOWER?"
26 GET A$ : IF A$ < > "H" AND A$ < > "L" THEN 25
27 PRINT "{HOME}{RIGHT}"; : IF I = 2
     THEN PRINT "{HOME}{11 RIGHT}";
28 IF I = 3
     THEN PRINT "{HOME}{13 RIGHT}
                 {9 RIGHT}";
29 IF I = 4
     THEN PRINT "{HOME}{15 RIGHT}
                 {16 RIGHT}";
30 PRINT "{6 DOWN}{UP}▒▒▒▒▒▒▒{DOWN}▮▮ ▮▮
     ▮▮ ▮▮ ▮▮ ▮▮ ▮▮ ▒ ▲ ▲ ▲ ▲ ▲ ▒{DOWN}▮▮ ▮▮ ▮▮ ▮▮ ▮▮ ▮▮ ▮▮ ▒ ▲
      ▲ ▲ ▲ ▲ ▒{DOWN}▮▮ ▮▮ ▮▮ ▮▮ ▮▮ ▮▮ ▮▮ ▒ ▲ ▲ ▲ ▲ ▲ ▒";
31 PRINT "{DOWN}▮▮ ▮▮ ▮▮ ▮▮ ▮▮ ▮▮ ▮▮ ▒ ▲ ▲ ▲ ▲ ▲ ▒";
32 PRINT "{DOWN}▮▮ ▮▮ ▮▮ ▮▮ ▮▮ ▮▮ ▮▮ ▒ ▲ ▲ ▲ ▲ ▲ ▒{DOWN}▮▮ ▮▮ ▮▮ ▮▮ ▮▮
     ▮▮ ▮▮ ▒ ▲ ▲ ▲ ▲ ▲ ▒{DOWN}▮▮ ▮▮ ▮▮ ▮▮ ▮▮ ▮▮ ▮▮ ▒ ▲ ▲ ▲ ▲
      ▲ ▒{DOWN}▮▮ ▮▮ ▮▮ ▮▮ ▮▮ ▮▮ ▮▮ ▒▒▒▒▒▒▒"
33 PRINT "{HOME}{RIGHT}";
     : IF I = 2 THEN
     PRINT "{HOME}{11 RIGHT}";
34 IF I = 3
     THEN PRINT "{HOME}{12 RIGHT}
                 {10 RIGHT}";
35 IF I = 4
```

```
   THEN PRINT "{S}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}
      {]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}";
37 PRINT "{Q}{Q}{Q}{Q}{Q}{Q}{Q}{]}{▓}";N$;
   : PRINT "{Q}{Q}{Q}";S$ ( P ) ; : PRINT "{Q}{Q}{Q}{▓}"
   ;N$
38 POKE 54276,33 : POKE 54273,6 : POKE 54272,206
   : FOR Z = 1 TO 200 : NEXT : POKE 54273,0
39 POKE 54272,0
40 IF A$ = "H" AND N1 > N THEN 80
41 IF A$ = "L" AND N1 < N THEN 80
50 FOR J = 1 TO 300 : NEXT J,I : FOR I = 1 TO 244
   STEP 2 : POKE 54273,I : NEXT : POKE 54273,0
51 IF D = 1 THEN PRINT "{S}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}
   {Q}{Q}{Q}{Q}{Q}{Q}{E}YOU'VE▲DOUBLED▲YOUR▲MONEY"
   : M = M * 2 : GOTO 53
52 PRINT "{S}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}
   {▓}WELL▲DONE▲YOU'VE▲WON▲5P" : M = M + 5
53 PRINT "{S}{E}MONEY▲▲▲▲▲■■■■■■■■■■";M;"P"
55 D = 0 : PRINT "{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}
   {Q}{Q}{Q}{Q}{Q}{Q}{Q}DO▲YOU▲WANT
   ▲TO▲PLAY▲AGAIN?"
56 GET A$ : IF A$ = "N" THEN END
57 IF A$ < > "Y" THEN 56
58 IF RND ( 1 ) < .5 THEN 70
59 PRINT "{Q}{▌}DO▲YOU▲WANT▲TO▲TRY▲'
   DOUBLE▲OR▲BROKE'?"
60 GET A$ : IF A$ = "N" THEN 70
63 IF A$ < > "Y" THEN 60
65 D = 1
70 IF CU > 9 THEN 4
75 GOTO 10
80 FOR I = 244 TO 1 STEP - 2 : POKE 54273,I : NEXT
   : POKE 54273,0
82 PRINT "{S}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}
   {▓}HARD▲LINES▲YOU'VE▲LOST!"
83 PRINT "{S}{E}MONEY▲▲▲▲▲■■■■■■■■■■";M;"P"
85 M = M - 5 : IF M = 0 OR D = 1 THEN PRINT "{♥}YOU'
   RE▲BROKE!▲BYE" : END
86 D = 0 : PRINT "DO▲YOU▲WANT▲TO▲PLAY▲AGAIN?
   ▲▲▲▲▲▲▲▲▲▲▲▲▲"
88 GET A$ : IF A$ = "Y" THEN 70
90 IF A$ < > "N" THEN 88
95 END
100 PRINT "{S}{Q}{Q}{Q}{Q}" : FOR I = 1 TO 19
105 PRINT "▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲
    ▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲";
106 NEXT
110 RETURN
```

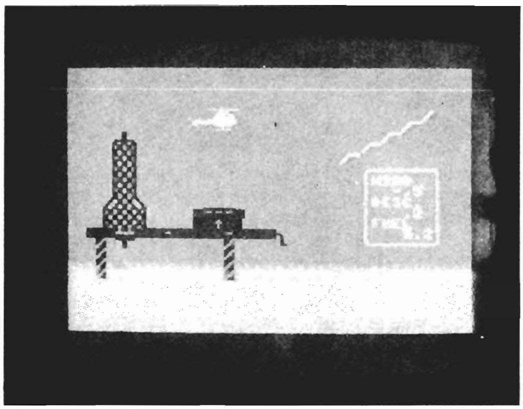
CHEXSUM

0=4148	55=3524
1=1958	56=1248
4=4331	57=1085
5=1963	58=1055
6=2088	59=3437
7=1973	60=1232
8=331	63=1080
10=1221	65=298
11=4369	70=757
13=1993	75=472
14=655	80=2694
15=2437	82=2907
16=1928	83=2126
17=1319	85=3848
18=1384	86=3194
19=1401	88=1245
20=1386	90=1060
21=1384	95=129
22=4816	100=1154
23=722	105=2034
24=2173	106=131
25=2173	110=143
26=2265	
27=1717	TOTAL= 136621
28=1594	
29=2125	
30=7509	
31=2008	
32=8410	
33=1717	
34=1594	
35=2125	
37=2582	
38=3762	
39=527	
40=1461	
41=1461	
50=3818	
51=5205	
52=3909	
53=2126	

SCREEN DISPLAY



North Sea Copter



This game really tests your skill and though it might seem otherwise it is possible to complete it.

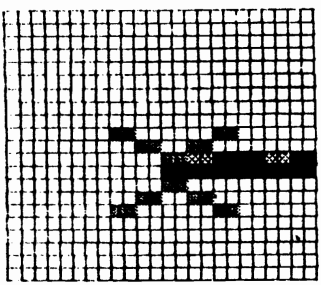
You must land your copter on an oil rig, at night, in the height of a storm, visibility is nil and you rely on lightning to see the rig. The wind buffets you about and as you manoeuvre or hover, gravity will pull you down, you must correct this by increasing lift.

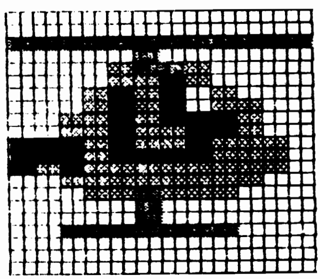
On the right of your screen are your instruments, watch your fuel and rate of descent, to land your rate must be below 2.

'A' moves you to the left, 'D' to the right and 'W' gives you lift.

PROGRAM STRUCTURE	LINES
INITIALIZATION	0—4
SETUP GRAPHICS	5—19
UPDATE GAUGES	20—21
LIGHTNING TIMING	22—35
READ KEYBOARD	36—55
UPDATE SPRITE POSITION	66—80
CRASH HANDLER	100—200
LANDING HANDLER	500—640
NOTE SEQUENCE FOR GOOD LANDING	800—890
SPRITE DATA	9000—9070

VARIABLES
V = POINTER TO DISPLAY CHIP LOCATIONS
CO = DIFFERENCE BETWEEN VIDEO AND SCREEN MEMORY
T = LIGHTNING TIMER
X = SPRITE X COORDINATE
Y = SPRITE Y COORDINATE
D = WIND DIRECTION
PE = SPRITE COLLISION FLAGS
F = FUEL LEFT
G = DESCENT RATE





SPRITE

SPRITES No. 1 & No. 2
FORM A
HELICOPTER

NORTH SEA COPTER

```
 0 PRINT "▣" : POKE 53280,0 : POKE 53281,0
     : V = 53248 : CO = 54272 : POKE 54296,10
     : POKE 54276,0
 1 POKE 54276,129 : POKE 54278,240
 2 POKE 54273,0 : T = 0 : X = 35 +
     INT ( RND ( 1 ) * 180 ) : Y = 55
     : D = ( INT ( RND ( 1 ) * 5 ) - 2 ) / 2
 3 F = 10 : POKE V + 21,3 : POKE 2040,13
     : POKE V + 39,1 : POKE 2041,14 : POKE V + 40,4
     : POKE V + 28,3
 4 POKE V + 37,5 : POKE V + 38,7 : FOR I = 0 TO 190
     : READ Q : POKE I + 832,Q : NEXT
 5 PRINT "▣▣▣▣▣▣▣▣▣▣▣▣▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮
     ▮▮▮▮▣▣▭▭▭▭▭▭▣▣▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▣
     ▣WIND ▲ ▲ ▣▣▣▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▣ ▲ ▲ ▲ ▲ ▲ ▲
     ▣"
 6 PRINT "▣▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▣ ▣DESC. ▲ ▣▣
     ▣▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▣ ▲ ▲ ▲ ▲ ▲ ▲ ▣"
 7 PRINT "▣▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▣ ▣FUEL ▲ ▲ ▣▣
     ▣▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▣ ▲ ▲ ▲ ▲ ▲ ▲ ▣▣▮▮▮▮▮▮▮▮▮▮
     ▮▮▮▮▮▮▣▭▭▭▭▭▭▣"
 8 FOR I = 56016 TO 56055 : POKE I,6
     : POKE I - CO,223 : NEXT
     : FOR I = 56056 TO 56296 : POKE I,6
 9 POKE I - CO,160 : NEXT : POKE V,X
     : POKE V + 2,X + 24 : POKE V + 1,Y
     : POKE V + 3,Y : PE = PEEK ( V + 31 )
10 PRINT "▣▣▣▣▣▣▣▣▣▣▣▣▣▣▣▣▣▮▮▮▮▮▮▣▣
     ▣▣▣▮▮▮▮▮▮▮▮▣▣▣▣▣▮▮▮▮▮▮▮▮▣▣▣▣▣
     ▮▮▮▮▮▮▮▮▣▣▣▣▣▮▮▮▮▮▮▮▮▣▣▣▣"
11 PRINT "▣▣▣▣▣▣▣▣▣▮▮▮▮▮▮▮▮▮▮▣▣▣▣▣▣
     ▲ ▲ ▲ ▲ ▣▣---▣▬▣▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮
     ▮▮▮▮▮▮▮▮▮▮▣▣▣▣▣▣ ▲ ▲ ▲ ▲ ▣ ▲ ▲↑ ▲ ▣"
15 PRINT "▣▣▣ ▲ ▲ ▲ ▣▣ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲
     ▲ ▬▣"
16 PRINT "▣▣▣▣▣▣▬▣▣▣▣▣▣▣▣▣▣▣▣▣▣
     ▣▣▣▣▬▣"
17 PRINT "▣▣▣▣▣▣▬ ▲ ▣▣▣▣▣▣▣▣▣▣▣▣"
18 PRINT "▣▣▣▣▣▣▬▣▣▣▣▣▣▣▣▣▣▣▣▣"
19 PRINT "▣▣▣▣▣▣▣▣▣▣▣▣▣▣▣▣▣▣▣▣"
20 PRINT "▣▣▣▣▣▣▣▣▣▣▣▣▣▣▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮
     ▮▮▣ ▲ ▲ ▲ ▮▮▮▮▮▮";D : PRINT "▣▣▮▮▮▮▮▮▮▮▮▮▮▮
     ▮▮ ▲ ▲ ▲ ▮▮▮▮▮▮▮▮"; INT ( G * 10 ) / 10
21 PRINT "▣▣▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮ ▲ ▲ ▲ ▲ ▮▮▮▮▮▮▮▮"
     ;INT ( F * 10 ) / 10
22 T = T + 1 : POKE 2041,14 : POKE 54273,100

25 IF T = 30 THEN POKE 53281,11
     : PRINT "▣▣▣▣▣▣▮▮▮▮▮▮▮▮▮▮▣▣▣▮▮▮▮▣▮▮
     ▮▮▣▣▮▮▮▮▣▮▮▮▮▣▣▮▮▮▮▣▮▮▮▮▣▣▮▮▮▮▣▮▮▮▮▣"

30 IF T = 32 THEN
     PRINT "▣▣▣▣▣▣▮▮▮▮▮▮▮▮▮▮ ▲ ▣▮▮▮▮ ▲ ▮▮▮▮ ▲
     ▣▮▮▮▮ ▲ ▮▮▮▮ ▲ ▣▮▮▮▮ ▲ ▮▮▮▮ ▲ ▣▮▮▮▮ ▲ ▮▮▮▮ ▲ "
```

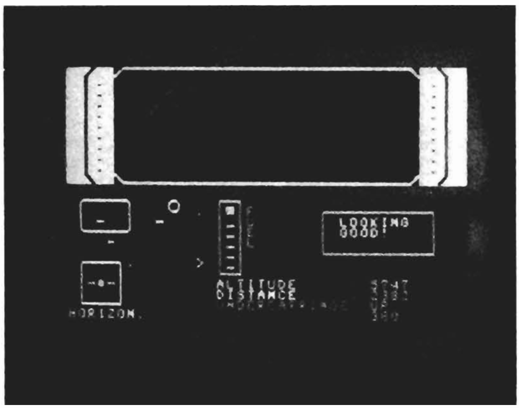
```
35 IF T = 35 THEN POKE 53281,0 : T = 0
36 POKE 54273,0
40 PE = PEEK ( 197 )
45 IF PE = 18 AND F > 0 THEN X = X + 2 : F = F - .1
50 IF PE = 10 AND F > 0 THEN X = X - 2 : F = F - .1
55 IF PE = 9 AND Y > 50 AND F > 0 THEN Y = Y - 2
     : G = 0 : F = F - .3
66 X = X + D : IF RND ( 1 ) < .1 THEN X = X + 2
     : F = F - .1
68 POKE 2041,15 : POKE 54273,100 : Y = Y + G
     : G = G + .2 : IF X > 215 THEN X = 215
69 IF X < 32 THEN X = 32
70 POKE V,X : POKE V + 1,Y : POKE V + 2,X + 24
     : POKE V + 3,Y
71 IF F < 0 THEN POKE 54296,1
72 IF X > 11 AND X < 120 AND Y > 135 THEN 500
78 IF PEEK ( V + 31 ) < > 0 THEN 100
79 POKE 54273,0
80 GOTO 20
100 POKE 53281,11
     : PRINT "▣ⓠⓠⓠⓠⓠ▮▮▮▮▮▮▮▮▮▮▣⊘ⓠ▮▮▮▮⋯▮▮
     ▮▮⊘ⓠ▮▮▮▮⋯▮▮▮▮⊘ⓠ▮▮▮▮⋯▮▮▮▮⊘ⓠ▮▮▮▮⋯▮▮▮▮⊘"
101 POKE 54296,15 : FOR I = 20 TO 0 STEP - .2
     : POKE 54273,I : NEXT : POKE 54273,0
110 FOR I = 1 TO 20 : POKE V,X - I : POKE V + 2,X + I
     : NEXT
120 FOR I = Y TO 220 : POKE V + 1,I : POKE V + 3,I
     : POKE 54273,I : NEXT : POKE 54273,0
130 POKE V,0 : POKE V + 1,0 : POKE V + 2,0
     : POKE V + 3,0
150 POKE 53281,0
     : PRINT "▣ⓠⓠⓠⓠⓠ▮▮▮▮▮▮▮▮▮▮▄⊘ⓠ▮▮▮▮⋯▮▮
     ▮▮⊘ⓠ▮▮▮▮⋯▮▮▮▮⊘ⓠ▮▮▮▮⋯▮▮▮▮⊘ⓠ▮▮▮▮⋯▮▮▮▮⊘"
180 FOR I = 1 TO 2000 : NEXT
200 RUN
500 IF G > 2 THEN 100
600 POKE 53281,2 : POKE V + 1,136 : POKE V + 3,136
     : POKE 54273,0 : POKE 54276,0 : POKE 54276,17
605 GOSUB 800
610 PRINT "▣ⓠⓠⓠⓠⓠⓠⓠⓠⓠⓠⓠⓠⓠⓠⓠⓠⓠⓠⓠⓠ
     ▣▣▣▣▣ ******* ▲WELL ▲DONE ▲*******"
620 PRINT "▣▣▣▣▣▣▣▣▣ⓠ ▲ ▲ ▲ANOTHER ▲GO? ▲ ▲ ▲"
625 GET A$ : IF A$ = "Y" THEN RUN
630 IF A$ < > "N" THEN 625
635 PRINT "▣▣▣BYE ▲BYE."
640 END
800 FOR I = 30 TO 100 STEP 2 : POKE 54273,I : NEXT
810 FOR I = 140 TO 95 STEP - 1 : POKE 54273,I : NEXT
820 FOR I = 5 TO 20 STEP .4 : POKE 54273,I : NEXT
830 FOR I = 90 TO 70 STEP - .3 : POKE 54273,I : NEXT
840 FOR I = 100 TO 0 STEP - 3 : POKE 54273,I : NEXT
850 FOR I = 20 TO 10 STEP - .2 : POKE 54273,I : NEXT
860 FOR I = 90 TO 70 STEP - .2 : POKE 54273,I : NEXT
870 FOR I = 0 TO 10 STEP .1 : POKE 54273,I : NEXT
880 POKE 54273,0 : POKE 54277,0
890 RETURN
```

```
9000 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
       0,0,0,0,0,0,0,64,64,0,17,0,0
9010 DATA 7,174,0,10,170,0,4,0,0,17,0,0,64,64,0,0,0,
       0,0,0,0,0,0,0,0,0,0,0,0,99
9020 DATA 0,0,0,0,0,0,85,85,85,0,16,0,0,215,0,0,251,
       0,3,184,192,3,184,240
9030 DATA 15,186,176,15,190,176,171,190,252,171,170,
       252,187,255,252,15,255,240
9040 DATA 3,223,192,0,16,0,0,16,0,5,85,64,0,0,0,0,0,
       0,0,0,0,99
9050 DATA 0,0,0,0,0,0,0,84,0,0,16,0,0,215,0,0,251,0,
       3,184,192,3,184,240
9060 DATA 15,186,176,15,190,176,171,190,252,171,170,
       252,187,255,252,15,255,240
9070 DATA 3,223,192,0,16,0,0,16,0,5,85,64,0,0,0,0,0,
       0,0,0,0
```

CHEXSUM

0=4299	68=4075	840=2037
1=1380	69=1083	850=2076
2=4814	70=2735	860=2083
3=4526	71=1159	870=1861
4=3580	72=2114	880=1167
5=9609	78=1582	890=143
6=5161	79=528	9000=4441
7=8739	80=472	9010=4295
8=4469	100=6918	9020=3704
9=5149	101=3420	9030=4321
10=3029	110=2603	9040=2987
11=9413	120=3698	9050=3510
15=1895	130=2279	9060=4321
16=2155	150=6966	9070=2802
17=1331	180=1020	
18=1320	200=139	TOTAL= 219038
19=1142	500=776	
20=6522	600=4027	
21=3433	605=302	
22=1905	610=2617	
25=7912	620=1733	
30=5338	625=1263	
35=1611	630=1125	
36=528	635=882	
40=796	640=129	
45=2580	800=1927	
50=2572	810=2091	
55=3527	820=1864	
66=2944	830=2084	

Flight Simulator



Test your skill at guiding your light aircraft onto the landing strip. Can you make a perfect landing? Watch your instruments as the runway comes up into view, you get all the feedback you need to do the job.

Your controls are:

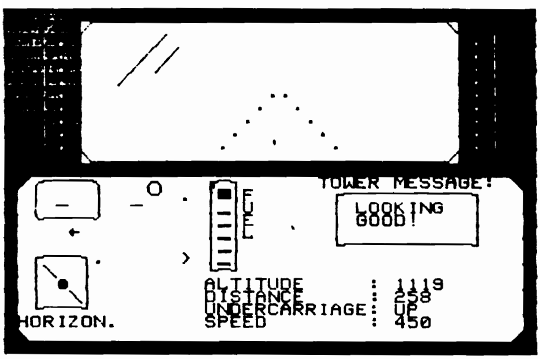
'F1' gives a cockpit display
'F3' gives a map
'A' tilts the plane to the left
'D' tilts it to the right
'W' alters levelling up
'X' alters levelling down
Keys 1 — 9 decide your acceleration/deceleration
'–' reverses the acceleration/deceleration rate
'space bar' lifts or lowers your undercarriage depending upon its current position.

There are many ways in which to 'fail' your landing:

- if the plane is tilted on an angle
- if the last altitude reading was above 50 (you've dropped the plane)
- if your attitude is not level
- if your undercarriage is not lowered

Safe landing!

SCREEN DISPLAY



PROGRAM STRUCTURE	LINES
INITIALIZATION	0 — 25
GET KEY PRESSED (A$)	30
PRINT COCKPIT DISPLAY	35
PRINT MAP DISPLAY	36
CHANGE ATTITUDE	50 — 56
RANDOM TILT	67
REVERSE ACCELERATION	70
READ SPEED KEY	75
LIFT OR LOWER UNDERCARRIAGE	80
CHANGE SPEED	90 — 95
CHANGE ALTITUDE	100 — 105
PRINT ALTITUDE, DISTANCE, SPEED	134 — 137
CHECK DISTANCE AND ALTITUDE	140 — 148
PRINT OUT COCKPIT MESSAGES	160 — 179
RESTART MAIN LOOP	180
SOUND ROUTINE	182 — 190
ROUTINES TO PRINT COCKPIT VIEWS	200 — 730
PRINT MAP	1000 — 1070
PRINT OUT DETAILS OF LANDING AND ASK IF PLAYER WANTS ANOTHER GAME	5000 — 6040

VARIABLES

AL = ALTITUDE
AT = ATTITUDE
U = UNDERCARRIAGE
DI = DISTANCE
A$ = HOLDS KEY PRESSED
CS = START OF COLOUR SCREEN MEMORY
SI = SPEED INCREMENT
SP = SPEED
SD = ACCELERATION OR DECELERATION
SC = SCREEN MEMORY
F = FUEL
H = TILT OF PLANE
M = 1 MAP VIEW, 0 COCKPIT VIEW
I = TEMPORARY STORAGE

```
0 PRINT "▣" : POKE 53281,0 : POKE 53280,9 :
    SC = 1024 : CS = 55296
1 POKE 54296,5 : POKE 54277,0 : POKE 54278,0 :
    POKE 54276,0 : POKE 54278,240 : POKE 54276,129
2 POKE 54273,0 : POKE 54272,0 : U = - 1 : AT = 0 :
    F = .6 : AL = 5000 + INT ( RND ( 1 ) * 1000 )
    : DI = 2500
3 H = 0 : M = 0 : SP = 300 : SI = 5 : SD = 1
5 PRINT "▣▣▣▲▲▣▲▲▬▣▣▢▢▢▢▢▢
    ▢▢▢▢▢▢▢▢▢▢▢▢▢▢▢▢▢▢▢▢
    ▢▢▣▣▲▲▣▲▲";
6 FOR I = 1 TO 9 : PRINT "▣▲▲▢.▲▬▣▲▣
    ▮▮▮▮▮▮▮▮▮▮▮▮▮▲▣▲.▢▲▲"; :
    NEXT
7 PRINT "▣▲▲▣▲▲▬▣▣▢▢▢▢▢▢▢▢▢
    ▢▢▢▢▢▢▢▢▢▢▢▢▢▢▢▢▢▢▢▣▣
    ▲▲▣▲▲"
8 PRINT "▣▣▣▣▣▣▣▣▣▣▣▣▣▣▣▣▣▮▮▣
    ▲▣▣▮▮▮▮▣▲▣▣▮▮▮▮▣"
10 FOR I = 440 TO 479 : POKE CS + I,9 :
    POKE SC + I,160 : NEXT
12 PRINT "▣▣▣▣▣▣▣▣▣▣▢▢▢▢▣▲▲▲▣
    ▣▲▲▲▬▣▢▣▲▲▲▲▲▲▣TOWER ▲MESSAGE:
    ▲▣";
14 PRINT "▣▲▢▲▲▲▲▲▢▲▲▲▣▣▲.▲▬▢▢▢
    ▣F▲▲▲▲▲▲▣▣▢▢▢▢▢▢▢▢▢▢▢▢▣"
15 PRINT "▣▲▢▲◣▢▣▲▲▢▲▲▢▲▲▲▲▲▲▬▢
    ▢▢▣U▲▲▲▲▲▲▣▢▲▲▲▲▲▲▲▲▲▲▲▢"
16 PRINT "▣▲▣▢▢▢▢▣▲▲▲▲▲▲▲▲▬▢▢
    ▢▣E▲▲▲▲▲▲▣▢▲▲▲▲▲▲▲▲▲▲▢"
17 PRINT "▲▲▲▲←▲▲▲▲▲▲▲▲▲▲▲▬▢▢▢▣
    L▲▲▲▲▲▲▣▢▲▲▲▲▲▲▲▲▲▲▢"
18 PRINT "▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲▬▢▢
    ▢▲▲▲▲▲▲▲▣▣▢▢▢▢▢▢▢▢▢▢▢▣"
19 PRINT "▣▲▣▢▢▢▣.▲▲▲▲▲▲>▲▬▢▢▢"
20 PRINT "▣▲▢▲▲▲▢▲▲▲▲▲▲▲▲▲▬▣▢▣"
21 PRINT "▣▲▢▬▢▣▬▢▣▢▲▲▲▲▲▲▲▲▲◣
    ALTITUDE▲▲▲▲▲:"
22 PRINT "▣▲▢▲▲▲▢▲▲▲▲▲▲▲▲▲DISTANCE
    ▲▲▲▲▲:"
23 PRINT "▣▲▣▢▢▢▣▲▲▲▲▲▲▲▲▲▲▣
    UNDERCARRIAGE:▲▣UP"
24 PRINT "▣HORIZON.▲▲▲▲▲▲▲▣SPEED
    ▲▲▲▲▲▲▲▲:"
25 POKE CS + 960,9 : POKE SC + 960,223 :
    POKE CS + 999,9 : POKE SC + 999,233
30 POKE 54273,0 : GET A$
35 IF A$ = "▬" AND M = 1 THEN PRINT "▣" : M = 0 :
    POKE 53281,0 : POKE 53280,9 : POKE 54273,100 :
    GOTO 5          Note: A$ = F1 Key
36 IF A$ = "▬" AND M = 0 THEN PRINT "▣" : M = 1 :
    GOSUB 1000  Note: A$   F3 Key
40 IF M = 1 THEN 90
50 IF A$ = "W" THEN AT = AT - 1 : IF AT < - 1 THEN
```

```
    AT = - 1
55 IF A$ = "X" THEN AT = AT + 1 : IF AT > 1 THEN
    AT = 1
56 IF A$ = "X" OR A$ = "W" THEN GOSUB 500
60 IF A$ = "A" THEN H = H - 1 : IF H < - 1 THEN
    H = - 1
65 IF A$ = "D" THEN H = H + 1 : IF H > 1 THEN H = 1
66 IF A$ = "A" OR A$ = "D" THEN GOSUB 600
67 IF RND ( 1 ) < .05 THEN H = INT
    ( RND ( 1 ) * 3 ) - 1 : GOSUB 600
70 IF A$ = "-" THEN SD = - SD : POKE 54273,1
74 POKE 54276,0
75 IF VAL ( A$ ) > 0 AND VAL ( A$ ) < 10 THEN
    SI = VAL ( A$ ) : POKE 54276,0 : POKE 54276,33 :
    POKE 54273,100
80 IF A$ = " ▲ " THEN U = - U : GOSUB 700
90 SP = SP + SI * SD : IF SP < 50 THEN SP = 50
95 IF SP > 450 THEN SP = 450
100 IF F < 6 THEN AL = AL - INT ( SP / 50 + AT *
    ( SP / 5 ) / 5 ) : GOTO 110
105 AL = AL - 100
110 IF F < 7 THEN F = F + ( SI / 3200 )
120 IF F < 7 THEN DI = DI - .1 -
    ( SP / 300 + U / 30 ) / 3 - SP / 50 : GOTO 130
125 DI = DI - .2
130 IF DI < 0 THEN 5000
132 IF DI > 30 AND AL < 0 THEN 5000
133 IF M = 1 THEN 140
134 PRINT "{CLR}{HOME}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}
    {DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}
    {RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}
    ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▌▌ ▌▌ ▌▌ ▌▌ ▌▌ ▌▌ ▌▌";AL
135 PRINT "{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}
    {RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT} ▲ ▲ ▲ ▲ ▲ ▲ ▌▌
    ▌▌ ▌▌ ▌▌ ▌▌ ▌▌ ▌▌"; INT ( DI )
136 PRINT "{DOWN}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}
    {RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT} ▲ ▲ ▲ ▲ ▲ ▲ ▌▌
    ▌▌ ▌▌ ▌▌ ▌▌ ▌▌ ▌▌";SP
137 PRINT "□□□□□□□□□□□{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}
    {RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{CLR}";
138 IF F < 7 THEN FOR I = 1 TO INT ( F ) : PRINT
    "{RVS}{ }{DOWN}▌▌"; : NEXT
139 PRINT
140 IF F > 5 THEN PRINT "{HOME}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}
    {DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RVS}{HOME}DANGER"
142 POKE 54273,0 : POKE 54276,0 : POKE 54276,129 :
    IF AL < 10 AND DI < 50 THEN POKE 54273,10
143 IF M = 1 THEN 30
144 IF INT ( DI ) > 600 AND INT ( DI ) < 700 THEN
    GOSUB 200
145 IF INT ( DI ) > 400 AND INT ( DI ) < 599 THEN
    GOSUB 250
146 IF INT ( DI ) > 200 AND INT ( DI ) < 299 THEN
    GOSUB 300
147 IF INT ( DI ) > 100 AND INT ( DI ) < 199 AND
    AL < 300 THEN GOSUB 350
```

```
148 IF INT ( DI ) < 80 AND INT ( DI ) > 30 AND
    AL < 200 THEN GOSUB 400
155 IF RND ( 1 ) < .8 THEN 30
156 PRINT "{S}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}■■
    ■■■■■■■■■■■■■■■■■■■■■■■■▲▲▲▲▲▲▲▲▲{Q}
    ■■■■■■■■■■■■■■■■■■■■▲▲▲▲▲▲▲▲▲"
157 PRINT "{S}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}
    ■■■■■■■■■■■■■■■■■■■■■■■■■■{E}";
160 IF RND ( 1 ) < .7 THEN 176
161 IF F > 4 THEN PRINT "▲LOW▲ON{Q}■■■■■■■■■■■■
    ▲FUEL" : GOTO 182
165 IF SP = 450 AND DI < 300 THEN PRINT "COMING
    ▲IN{Q}■■■■■■■■■■■■■■■■■■■■■■TOO▲FAST" :
    GOTO 182
170 IF AL < 500 AND DI > 500 THEN PRINT "YOUR▲TOO
    ▲{Q}■■■■■■■■■■■■■■■■■■■■▲LOW!" : GOTO 182
175 IF F > 3 AND SP > 300 THEN PRINT
    "{Q}■■TOO▲FAST!" : GOTO 182
176 IF F > 4 OR SP = 450 AND DI < 300 OR AL < 500
    AND DI > 500 OR F > 3 AND SP > 300 OR
    RND ( 1 ) > .2 THEN 30
178 PRINT "LOOKING▲▲{Q}■■■■■■■■■■■■■■■■■■■■▲GOOD'"
179 POKE 54276,0 : POKE 54276,17 : FOR I = 1 TO 30 :
    POKE 54273,I : NEXT : POKE 54273,0
180 GOTO 30
182 POKE 54276,0 : POKE 54276,33
184 FOR I = 10 TO 20 STEP .4 : POKE 54273,I : NEXT :
    POKE 54273,50
185 FOR I = 1 TO 100 : NEXT
190 GOTO 30
200 PRINT "{S}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{]}{]}{]}{]}{]}{]}{]}{]}
    {]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{π}..{Q}■■■■■■.▲▲."
210 RETURN
250 PRINT "{S}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{]}{]}{]}{]}{]}{]}{]}{]}{]}
    {]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{π}..{Q}■■■■■■.▲▲.{Q}
    ■■■■■■■■■■.▲▲▲▲."
260 RETURN
300 PRINT "{S}{Q}{Q}{Q}{Q}{Q}{]}{]}{]}{]}{]}{]}{]}{]}{]}
    {]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}";
305 PRINT "{π}..{Q}■■■■■■.▲▲.{Q}■■■■■■■■■■.▲▲▲▲.
    {Q}■■■■■■■■■■■■■■.▲▲▲▲▲▲.{Q}■■■■
    ■■■■■■■■■■■■■■.▲▲▲▲▲▲▲▲."
310 RETURN
350 PRINT "{S}{Q}{Q}{Q}{Q}{Q}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}
    {]}{]}{]}{]}{]}{]}{]}{]}{]}";
355 PRINT "{π}...{Q}■■■■■■■■.....{Q}■■■■■■■■■■■■.
    ▲▲.▲▲.{Q}■■■■■■■■■■■■■■■■.▲▲▲.▲▲▲.";
356 PRINT "{Q}■■■■■■■■■■■■■■■■■■■■.▲▲▲▲.
    ▲▲▲▲."
360 RETURN
400 PRINT "{S}{Q}{Q}{Q}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}
    {]}{]}{]}{]}{]}{]}{]}{]}";
405 PRINT "{π}◲▲◱▲{Q}■■■■■■■■■■◲...◱{Q}■■■■■■
    ■■■■■■◲▲▲.▲▲◱{Q}■■■■■■■■■■■■■■■■◲
    ▲▲▲.▲▲▲◱";
410 PRINT "{Q}■■■■■■■■■■■■■■■■■■■■◲▲▲▲▲.
```

```
      ▲▲▲▲◳Ⓠ▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮◳
      ▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲◳Ⓠ▮▮▮▮▮▮
      ▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮";
 411 PRINT "▩▲▲▲▲▲▲▲▲▲▲▲▲▲▲▩"
 420 RETURN
 500 PRINT "⊟ⓆⓆⓆⓆⓆⓆⓆⓆⓆⓆⓆⓆⓆ" : IF AT = - 1
      THEN PRINT "◘◘◘◣◹"
 510 IF AT = 0 THEN PRINT "◘◘◘◣⊖"
 530 IF AT = 1 THEN PRINT "◘◘◘◣◸"
 535 POKE 54276,0 : POKE 54276,129 : POKE 54273,20
 540 RETURN
 600 PRINT "⊟◘◘ⓆⓆⓆⓆⓆⓆⓆⓆⓆⓆⓆⓆⓆⓆ
      ⓆⓆⓆⓆ▲▲▲Ⓠ▮▮▮▮▮▮▲◘▲Ⓠ▮▮▮▮▮▮▲▲▲▩"
 605 IF H = 1 THEN PRINT "◻◻◻◘◘◹ⓆⓆ◘◹"
 610 IF H = - 1 THEN PRINT "◻◻◻◘◘◘◘◸ⓆⓆ
      ▮▮▮▮▮▮◸"
 615 IF H = 0 THEN PRINT "◻◻◘◘⊖◘⊖"
 616 POKE 54276,0 : POKE 54276,129 : POKE 54273,40
 620 RETURN
 700 PRINT "⊟ⓆⓆⓆⓆⓆⓆⓆⓆⓆⓆⓆⓆⓆⓆⓆⓆ
      ⓆⓆⓆⓆⓆⓆ◘◘◘◘◘◘◘◘◘◘◘◘◘◘◘◘
      ◘◘◘◘◘◘◘◘◘◘◘◘◘◘◘◘◘";
 710 IF U = - 1 THEN PRINT "▥UP▲▲"
 720 IF U = 1 THEN PRINT "▨DOWN"
 725 POKE 54276,0 : POKE 54276,33 : POKE 54273,5
 730 RETURN
1000 POKE 53281,5 : POKE 53280,0
1005 PRINT "⊟⊟▲▲▲▲▲MAP:"
1010 PRINT "ⓆⓆⓆ◘◘◘◘◘◘◘▲*▲⊟⊖CONTROL
      ▲TOWER."
1020 PRINT "◘◘◘◘◘◘◘▨◙◨◧"
1030 PRINT "◘◘◘◘◘◘◘◙◨◧⊟▬⊖RUNWAY"
1050 PRINT "▙ⓆⓆⓆⓆⓆⓆⓆⓆⓆⓆⓆⓆⓆⓆ◘◘◘
      ◘◘◘◘▩⊟⊖STARTING▲POINT"
1060 PO = 207 + INT ( DI / 150 ) * 40 : POKE CS +
      PO,0 : POKE SC + PO,160
1070 RETURN
5000 POKE 53280,0 : POKE 53281,0 : PRINT
      "♥ⓆⓆⓆⓆⓆ▧"
5010 IF AL > 50 THEN PRINT "YOU▲MISSED▲THE
      ▲RUNWAY!" : GOTO 6000
5020 IF AL < 0 THEN PRINT "←←YOU▲CRASHED←←" :
      GOTO 6000
5030 IF H = - 1 AND RND ( 1 ) < .7 THEN PRINT
      "YOUR▲LEFT▲WING▲HIT▲THE▲RUNWAY!" :
      GOTO 6000
5040 IF H = 1 AND RND ( 1 ) < .7 THEN PRINT
      "YOUR▲RIGHT▲WING▲HIT▲THE▲RUNWAY!" :
      GOTO 6000
5050 IF AT = 1 THEN PRINT "**YOU▲NOSE-DIVED!**" :
      GOTO 6000
5055 IF AT = - 1 THEN PRINT "**YOU▲BELLY-FLOPPED
      !**" : GOTO 6000
5060 IF SP > 95 THEN PRINT "↑↑YOU▲CAME▲IN▲TOO
      ▲FAST!↑↑" : GOTO 6000
5065 IF U = - 1 THEN PRINT "YOU▲DIDN'T▲LOWER▲YOUR
```

```
        ▲UNDERCARRIAGE."
5070 IF F > 5 OR SP > 90 THEN PRINT "YOU▲JUST▲MADE
        ▲IT!" : GOTO 6005
5080 IF SP > 70 THEN PRINT "**A▲HARD▲LANDING!**" :
       GOTO 6005
5090 IF SP > 60 THEN PRINT "▒▒A▲GOOD▲LANDING
       !▒▒" : GOTO 6005
5100 PRINT "◣THAT▲WAS▲A▲PERFECT▲LANDING!"
5105 POKE 54276,0 : POKE 54276,17 : FOR I = 1 TO 100
       STEP 2 : POKE 54273,I : NEXT
5106 POKE 54273,0
5110 PRINT "▣▣BRILLIANT!▣▲SUPER!" : GOTO 6005
6000 POKE 54296,15
6001 POKE 54276,0 : POKE 54276,129 : FOR I = 1 TO 10
       STEP .2 : POKE 54273,I : NEXT
6002 FOR I = 30 TO 0 STEP - .5 : POKE 54273,I : NEXT
6003 FOR I = 1 TO 300 : NEXT : POKE 54273,0
6005 PRINT "▣▣▣▣■DO▲YOU▲WANT▲ANOTHER▲GO?"
6010 GET A$ : IF A$ = "Y" THEN RUN
6020 IF A$ < > "N" THEN GOTO 6010
6030 PRINT "▣BYE."
6040 END
```

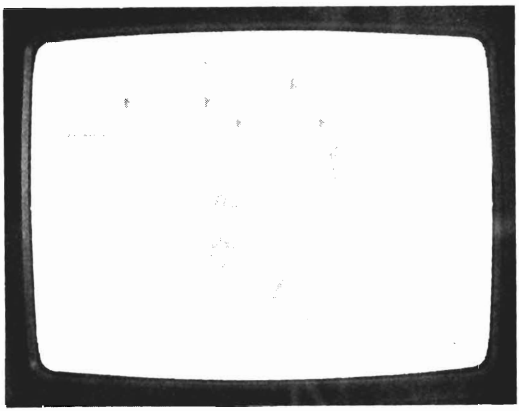
CHEXSUM

0=3002	36=2557	135=3310
1=3899	40=735	136=3051
2=5296	50=3140	137=2624
3=2233	55=2760	138=2586
5=7026	56=1762	139=153
6=4190	60=2737	140=2416
7=6856	65=2353	142=4114
8=3898	66=1723	143=727
10=2795	67=2793	144=2387
12=5563	70=2030	145=2407
14=5690	74=527	146=2395
15=4113	75=4753	147=3143
16=4179	80=1627	148=2997
17=3186	90=2518	155=1046
18=4880	95=1375	156=5476
19=2468	100=4019	157=2757
20=2012	105=811	160=1119
21=3259	110=1777	161=3399
22=2321	120=4317	165=6061
23=3388	125=748	170=5369
24=2151	130=913	175=3161
25=3623	132=1554	176=5700
30=877	133=783	178=2992
35=4638	134=4655	179=3614

```
180=472
182=1219
184=2619
185=964
190=472
200=2118
210=143
250=3406
260=143
300=1045
305=6203
310=143
350=1045
355=4857
356=2422
360=143
400=993
405=6167
410=8606
411=1203
420=143
500=2285
510=1399
530=1399
535=1964
540=143
600=2497
605=1898
610=2579
615=1667
616=1962
620=143
700=2515
710=1481
720=1227
725=1838
730=143
1000=1176
1005=761
1010=2084
1020=878
1030=1760
1050=2762
1060=3668
1070=143
5000=1804
5010=3454
5020=2939
5030=5329
5040=5235
5050=3056
5055=3516
5060=3840
5065=4263
5070=3603
5080=3085
5090=3623
5100=2574
5105=3249
5106=528
5110=2449
6000=595
6001=3342
6002=2033
6003=1587
6005=2352
6010=1263
6020=1570
6030=517
6040=129

TOTAL= 366336
```

Slalom



Can you complete the slope? If you can, how many points can you get?

You gain points for passing through the posts and crash if you hit a tree or an ice crack.

Watch the skier jump for joy if he completes the course or end up, upside-down in the snow if he trips on a tree!

Your controls are:
'Z' tilts the skis left
'M' tilts the skis right

Movement continues in the chosen direction until changed!

PROGRAM STRUCTURE	LINES
INITIALIZATION Note: 6 clears collision register by PEEKing it	0 — 6
FINISHED COURSE?	10
READ KEY If course ending print finish line	11
GO LEFT?	12
GO RIGHT?	14
CHECK FOR LEFT, RIGHT BOUNDARIES	15 — 16
UPDATE POSITION OF SKIER	17 — 18
HIT ANYTHING?	19
PRINT AND MOVE TREES	20 — 80
FINISHED COURSE	100
WHAT HAS BEEN HIT?	101 — 108
SOUND ROUTINE	200 — 220
PRINT FINISH LINE ROUTINE	300 — 310
FINISHED ROUTINE WITH JUMPING SKIER	500 — 540
ANOTHER GAME?	550 — 580
MISSED FINISH POSTS	1000 — 1050
DATA	9000 — 10040

VARIABLES
D = DIRECTION OF SKIER: 0 = LEFT, 1 = STRAIGHT, 2 = RIGHT
V = VIDEO CHIP
SL = LENGTH OF COURSE COVERED
S = SCORE
I, Q, A, Z = TEMPORARY STORAGE
X = POSITION OF SKIER (HORIZONTAL)
PE = KEY PRESSED
X1 = TREE POSITION
T = NO. OF TREES

SLALOM

```
 0 POKE 55,255 : POKE 56,47 : POKE 53281,15 :
     POKE 53280,15 : PRINT "♥"
 1 D = 1 : X = 140 : V = 53248 : SL = 0 : S = 0 :
     POKE V + 21,1
 2 POKE 2040,193 : POKE V + 39,2 :
     FOR I = 0 TO 318 : READ Q :
     POKE 12288 + I,Q : NEXT
 3 POKE 54278,0 : POKE 54276,0 : POKE 54273,0 :
     POKE 54272,0
 4 POKE 54296,10 : POKE 54278,240 : POKE 54276,129
 5 FOR I = 12800 TO 12871 : READ A : POKE I,A :
     NEXT : POKE V + 24,28 : FOR I = 12544 TO 12551
 6 POKE I,0 : NEXT : POKE V,X : POKE V + 1,60 :
     Z = PEEK ( V + 31 )
10 S = S + 1 : SL = SL + 1 : PRINT "[gfx][gfx][gfx][gfx][gfx][gfx][gfx]
     [gfx][gfx][gfx][gfx][gfx][gfx][gfx][gfx][gfx][gfx][gfx][gfx][gfx][gfx][gfx][gfx][gfx]" :
     IF SL = 220 THEN 1000
11 POKE 54273,0 : PE = PEEK ( 197 ) :
     IF SL = 180 THEN GOSUB 300
12 IF PE = 12 AND D > 0 THEN D = D - 1 :
     POKE 2040,192 + D : POKE 54273,100
14 IF PE = 36 AND D < 2 THEN D = D + 1 :
     POKE 2040,192 + D : POKE 54273,100
15 IF X < 45 THEN D = 1 : X = 45 :
     POKE 2040,192 + D : GOTO 18
16 IF X > 239 THEN D = 1 : X = 239 :
     POKE 2040,192 + D : GOTO 18
17 X = X + ( D - 1 ) * 3
18 POKE V,X
19 IF PEEK ( V + 31 ) = 1 THEN 100
20 IF RND ( 1 ) < .7 AND T = 0 AND SL < 175 THEN
     T = 1 : X1 = INT ( RND ( 1 ) * 30 )
25 IF PE = 12 OR PE = 36 THEN POKE 54273,0
30 IF RND ( 1 ) < .1 AND T = 0 AND SL < 175 THEN
     PRINT TAB( INT ( RND ( 1 ) * 30 ) )
     ;"[gfx][gfx][gfx]AAAAAAA[gfx][gfx]"
31 IF RND ( 1 ) < .1 AND T = 0 AND SL < 175 THEN
     PRINT TAB( INT ( RND ( 1 ) * 30 ) )
     ;"[gfx]♠♠♠♠" : GOTO 10
40 IF T = 0 THEN PRINT : GOTO 10
50 IF T = 1 THEN PRINT TAB( X1 ) ;"[gfx][gfx][gfx][gfx]" :
     T = 2 : GOTO 10
60 IF T = 2 THEN PRINT TAB( X1 ) "[gfx][gfx][gfx][gfx][gfx]" :
     T = 3 : GOTO 10
70 IF T = 3 THEN PRINT TAB( X1 ) "[gfx][gfx][gfx][gfx][gfx]" :
     T = 4 : GOTO 10
80 IF T = 4 THEN PRINT TAB( X1 ) "[gfx][gfx][gfx]"  :
     T = 0 : GOTO 10
100 IF SL > 197 THEN 500
101 IF PEEK ( 1104 + ( X - 20 ) / 8 ) = 1 OR
     PEEK ( 1405 + ( X - 20 ) / 8 ) = 1 THEN
     GOTO 20
102 IF PEEK ( 1106 + ( X - 20 ) / 8 ) = 1 THEN
     GOTO 20
```

```
103 IF PEEK ( 1144 + ( X - 20 ) / 8 ) = 1 OR
      PEEK ( 1145 + ( X - 20 ) / 8 ) = 1 THEN
      GOTO 200
104 IF PEEK ( 1146 + ( X - 20 ) / 8 ) = 1 THEN
      GOTO 200
105 IF PEEK ( 1064 + ( X - 20 ) / 8 ) = 1 OR
      PEEK ( 1065 + ( X - 20 ) / 8 ) = 1 THEN
      GOTO 20
106 IF PEEK ( 1066 + ( X - 20 ) / 8 ) = 1 THEN
      GOTO 20
107 IF PEEK ( 1024 + ( X - 20 ) / 8 ) = 1 OR
      PEEK ( 1025 + ( X - 20 ) / 8 ) = 1 THEN
      GOTO 20
108 IF PEEK ( 1026 + ( X - 20 ) / 8 ) = 1 THEN
      GOTO 200
110 POKE 54296,15
115 POKE 2040,196 : FOR I = 60 TO 170 :
      POKE 54273,175 - I : POKE V + 1,I : NEXT
116 POKE 54273,1
120 POKE 2040,195
130 FOR I = 1 TO 700 : NEXT : POKE 54273,0
140 POKE V,0 : POKE V + 1,0 : POKE 53280,0 :
      POKE 53281,0 : POKE V + 24,20
150 PRINT "■ ■HARD LUCK YOU'VE CRASHED!"
160 PRINT "■ ■ ■ ■ ◣YOU COMPLETED ■"
      INT ( SL / 2 ) ;" % ◣ OF THE"
170 PRINT "COURSE. ■YOU SCORED "
      INT ( S / 4 ) "POINTS."
180 GOTO 550
200 S = S * 2 : POKE 54296,15 : POKE 54276,0 :
      POKE 54276,33 : FOR I = 20 TO 50 :
      POKE 54273,I : NEXT
210 POKE 54296,15 : POKE 54276,0 : POKE 54276,129 :
      POKE 54273,0
220 GOTO 20
300 PRINT "■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ▒ ▤ ▢
      ▌ ▤ ▢ ▌ ▤ ▤ ▤ ▤ ▤ ▤ ▤ ▤ ▤ ▒ ▤ ■ ▌ ▤ ■ ▌ ▤"
310 RETURN
500 POKE 54273,10 : POKE 2040,193 :
      FOR I = 1 TO 5 : FOR J = 1 TO 20 STEP 1 :
      FOR Z = 1 TO 20 : NEXT
510 POKE V + 1,50 + J : NEXT J : FOR J = 20 TO 1
      STEP - 1 : FOR Z = 1 TO 20 : NEXT :
      POKE V + 1,50 + J : NEXT J,I
515 FOR I = 1 TO 240 STEP 5 : POKE 54273,I : NEXT :
      POKE 54273,0
520 POKE V,0 : POKE V + 1,0 : POKE 53280,0 :
      POKE 53281,0 : POKE V + 24,20
530 PRINT "■ ■YOU'VE COMPLETED THE COURSE!"
540 PRINT "■ ■ ▒YOU SCORED ◣"; S ;
      " ▒ POINTS."
550 PRINT "■ ■ ■DO YOU WANT TO PLAY AGAIN?"
560 GET A$ : IF A$ = "Y" THEN RUN
```

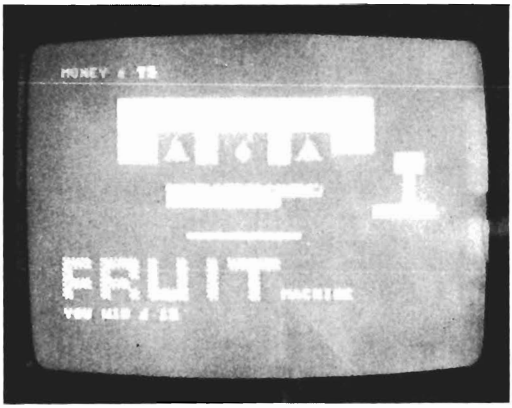
```
 570 IF A$ < > "N" THEN 560
 575 PRINT "BYE"
 580 END
 1000 FOR I = 1 TO 240 : POKE 54273,I : NEXT :
        POKE 54273,0 : POKE 53280,0 : POKE 53281,0
 1010 POKE V,0 : POKE V + 1,0 : POKE V + 24,20 :
        PRINT "HA HA!"
 1020 PRINT "YOU MISSED THE FINISH POST!"
 1030 PRINT "BAD LUCK."
 1040 PRINT "YOU SCORED ";INT ( S/4 ) ; " POINTS."
 1050 GOTO 550
 9000 DATA 0,56,0,0,56,0,0,56,0,0,16,0,0,124
        ,0,0,187,0,1,57,0,1,57,0,0,185
 9010 DATA 0,0,165,0,0,36,0,0,68,0,0,66,0,0
        ,34,0,0,35,0,0,50,0,0,36,0,0,72,0
 9020 DATA 0,144,0,1,32,0,2,64,0,99
 9030 DATA 0,56,0,0,56,0,0,56,0,0,16,0,0,124
        ,0,0,187,0,1,57,0,1,57,0,0,185
 9040 DATA 0,0,165,0,0,36,0,0,68,0,0,66,0
        ,0,42,0,0,38,0,0,50,0,0,33,0,0,64,128
 9050 DATA 0,128,64,1,0,0,2,0,0,99
 9060 DATA 0,56,0,0,56,0,0,56,0,0,16,0,0
        ,124,0,0,187,0,1,57,0,1,57,0,0,185
 9070 DATA 0,0,165,0,0,36,0,0,68,0,0,66,0
        ,0,42,0,0,38,0,0,98,0,0,33,0,0,16,128
 9080 DATA 0,8,64,0,4,32,0,2,16,99
 9100 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
        ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
 9110 DATA 0,0,0,16,0,8,8,0,16,4,0,32,2,0
        ,64,1,0,128,0,129,0,2,66,64
 9120 DATA 0,0,0,0,255,0,99
 9160 DATA 2,0,0,1,0,0,0,128,64
 9170 DATA 0,64,128,0,33,0,0,50,0,0,38,0,0
        ,42,0,0,66,0,0,68,0,0,36,0,0,165,0
 9180 DATA 0,185,0,1,57,0,1,57,0,0,187,0,0
        ,124,0,0,16,0,0,56,0,0,56,0,0,56,0
10000 DATA 255,255,255,255,255,255,255,255
10010 DATA 0,0,0,59,255,231,0,0,128,192,224
        ,240,128,128,128,128,3,3,3,3,3,3,3,3
10020 DATA 192,192,192,192,192,192,192,192
10030 DATA 0,0,24,24,60,126,126,255,3,3,7,15
        ,31,63,63,127,192,192,224,240,240
10040 DATA 248,252,252,60,60,60,60,60,60,60,60
```

CHEXSUM

```
0=2830
1=3026
2=3673
3=2383
4=2089
5=4182
6=2753
10=3754
11=2723
12=3453
14=3457
15=2994
16=3110
17=1153
18=376
19=1393
20=3777
25=1955
30=5392
31=5193
40=1339
50=2568
60=2907
70=2917
80=2457
100=1006
101=4334
102=2449
103=4395
104=2506
105=4336
106=2449
107=4332
108=2506
110=595
115=3526
116=529
120=576
130=1587
140=2981
150=2402
160=3334
170=3119
180=530
200=4422
210=2552
220=472
300=5050
310=143
500=4251
510=5001
515=2516
520=2981
530=2644
540=2579
550=2476
560=1263
570=1129
575=543
580=129
1000=3519
1010=2716
1020=2593
1030=906
1040=2786
1050=530
9000=3698
9010=3934
9020=1449
9030=3698
9040=4174
9050=1406
9060=3698
9070=4198
9080=1380
9100=3861
9110=3293
9120=977
9160=1175
9170=3913
9180=3949
10000=1886
10010=4281
10020=1902
10030=4066
10040=2100

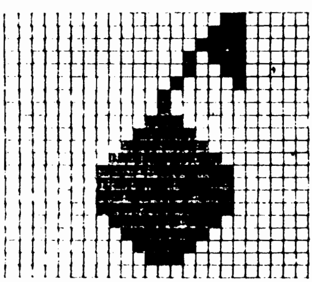
TOTAL= 228575
```

Fruit Machine



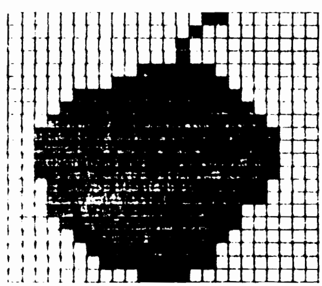
A version of the old favourite. It costs you £10 a go and you can win up to £100. Pull the handle and try your luck!

The program contains some nice graphics. All instructions are included.



SPRITE

CHERRY



SPRITE

STRAWBERRY

PROGRAM STRUCTURE	LINES
SPRITE INITIALIZATION	1 — 10
DRAW MACHINE	20 — 115
GET KEY: CONTINUE OR END	120
CALL 'HANDLE DOWN'	122
GET 3 RANDOM FRUITS	125 — 140
POSITION FRUITS	141 — 142
CALCULATE PRIZE	160 — 170
DISPLAY FRUITS	171 — 175
DRAW HANDLE UP	178 — 180
CALCULATE WALLET	190 — 200
IF BROKE, CONTINUE OR END	210 — 5000
DRAW 'HANDLE DOWN'	8000 — 8030
SPRITE DATA	9000 — 9080

VARIABLES
M = MONEY
W = PRIZE MONEY FOR CURRENT SPIN
V = SPRITE POINTER
R(X) = FRUIT ARRAY (X = 1, 2, or 3)

FRUIT MACHINE

```
1 M = 30 : W = 0
2 DIM R ( 3 ) : V = 53248 : POKE V + 21,7
3 POKE 2040,13 : POKE 2041,14 : POKE 2042,15
   : POKE V + 39,2 : POKE V + 40,5 : POKE V + 41,9
4 FOR N = 0 TO 190 : READ Q : POKE 832 + N,Q : NEXT
5 POKE V,108 : POKE V + 1,100 : POKE V + 2,160
   : POKE V + 3,100 : POKE V + 4,210
   : POKE V + 5,100
10 POKE 53281,0 : POKE 53280,0
20 PRINT "{CLR}{WHT}MONEY £"
30 PRINT "{DOWN}{DOWN}{RIGHT}      {RVS}                              "
31 PRINT "{RIGHT}      {RVS}                            "
32 PRINT "{RIGHT}      {RVS}                            "
33 PRINT "{RIGHT}      {RVS}    {OFF}    {RVS}  {OFF}     {RVS}  {OFF}     {RVS}     "
34 PRINT "{RIGHT}      {RVS}    {OFF}    {RVS}  {OFF}     {RVS}  {OFF}     {RVS}     "
35 PRINT "{RIGHT}      {RVS}    {OFF}    {RVS}  {OFF}     {RVS}  {OFF}     {RVS}     "
36 FOR I = 1 TO 7
37 PRINT "{RIGHT}      {RVS}                            "
38 NEXT
39 PRINT "{HOME}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{LEFT}{LEFT}{LEFT}{LEFT}{LEFT}{LEFT}{LEFT}{PUR}{RVS}   {OFF}{DOWN}{LEFT}{LEFT}{LEFT}{RVS}    {OFF}{DOWN}{LEFT}{LEFT}{UP}{RVS} {DOWN}{LEFT} {DOWN}{LEFT} {DOWN}{LEFT}{LEFT}{LEFT}{LEFT}{LEFT}    "
42 PRINT "{YEL}{HOME}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}■■■ ■■■■ ■  ■  ■ ■■■■■"
43 PRINT "■   ■  ■ ■  ■  ■   ■"
44 PRINT "■■■ ■■■  ■  ■  ■   ■"
45 PRINT "■   ■  ■ ■■■■  ■   ■  MACHINE"
46 PRINT "{DOWN}{RED}YOU WIN £"
100 PRINT "{HOME}{UP}{UP}{UP}{UP}{UP}{UP}{UP}{UP}     {LEFT}{LEFT}{LEFT}{LEFT}{LEFT}";M
110 PRINT "{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{UP}{UP}{UP}{UP}{UP}{UP}{UP}{UP}{UP}{UP}{UP}{RIGHT}'C' TO CARRY ON{DOWN}{LEFT}{LEFT}{LEFT}{LEFT}{LEFT}{LEFT}{LEFT}{LEFT}{LEFT}{LEFT}{LEFT}{LEFT}{LEFT}'Q' TO QUIT"
115 POKE 198,0
120 GET A$ : IF A$ = "Q" THEN 5000
121 IF A$ < > "C" THEN 120
122 GOSUB 8000
125 FOR I = 1 TO 3
130 R ( I ) = INT ( RND ( 1 ) * 3 + 1 )
   : IF RND ( 1 ) < .7 AND R ( I ) = R ( I - 1 )
   THEN 130
131 IF R ( I ) = 3 AND RND ( 1 ) < .6 THEN 130
140 NEXT
141 IF I = 2 THEN PRINT "{HOME}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{UP}{UP}{UP}
```

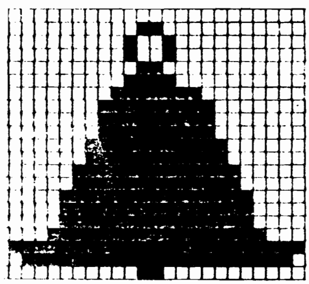
```
      []][]][]][]][]][]][]][]][]][]][]][]][]]";
 142 IF I = 3 THEN PRINT "[S][Q][Q][Q][Q][Q][Q][Q][]][]][]][]][]]
      []][]][]][]][]][]][]][]][]][]][]][]][]][]][]][]][]][]]";
 160 IF R ( 1 ) = 1 AND R ( 2 ) = 1 AND R ( 3 ) = 1
      THEN W = 10
 161 IF R ( 1 ) = 1 AND R ( 2 ) = 1 AND R ( 3 ) < > 1
      THEN W = 5
 162 IF R ( 1 ) = 2 AND R ( 2 ) = 2 AND R ( 3 ) = 3
      THEN W = 20
 163 IF R ( 1 ) < > 2 AND R ( 2 ) = 2
      AND R ( 3 ) < > 2 THEN W = 15
 164 IF R ( 1 ) = 3 AND R ( 2 ) = 3 AND R ( 3 ) = 3
      THEN W = 100
 165 IF R ( 1 ) = 3 AND R ( 2 ) = 3 AND R ( 3 ) < > 3
      THEN W = 50
 166 IF R ( 1 ) < > 3 AND R ( 2 ) = 3
      AND R ( 3 ) < > 3 THEN W = 50
 170 IF W < > 0 AND RND ( 1 ) < .8 THEN 125
 171 POKE V + 21,7 : FOR Z = 1 TO 3
 172 IF R ( Z ) = 1 THEN POKE V + 38 + Z,2
 173 IF R ( Z ) = 2 THEN POKE V + 38 + Z,5
 174 IF R ( Z ) = 3 THEN POKE V + 38 + Z,9
 175 POKE 2039 + Z,R ( Z ) + 12 : NEXT
 178 PRINT "[S][Q][Q][Q][Q][Q][Q][Q][Q][Q][Q][Q][Q][Q][Q]
      █████████▲▲▲▲[Q]█▲[Q]█▲[Q]█▲[Q]
      ██▲▲▲[Q]███▲▲▲"
 179 PRINT "[S][Q][Q][Q][Q][Q][Q][Q][Q][Q]███████[E][R]▲
      ▲▲[-][Q]███[R]▲▲▲[-][Q]██[E][R]▲[Q]█▲[Q]
      █▲[Q]████▲▲▲▲"
 180 PRINT "[S][Q][Q][Q][Q][Q][Q][Q][Q][Q][Q][Q][Q][Q][Q][Q][Q][Q]
      [Q][Q][Q][Q][Q][Q][]][]][]][]][]][]][]][]][]]▲▲▲███"W
 190 M = M - 10
 200 M = M + W : W = 0 : IF M > 0 THEN GOTO 100
 210 PRINT "[S][E]YOUR▲BROKE!!"
 220 PRINT "[E]WANT▲ANOTHER▲GAME?"
 230 GET A$ : IF A$ = "Y" THEN RUN
 240 IF A$ < > "N" THEN 230
 250 END
5000 PRINT "[C]" : PRINT "BYE▲BYE▲THEN"
5010 END
8000 PRINT "[S][Q][Q][Q][Q][Q][Q][Q][Q][Q]███████▲▲
      ▲[Q]███▲▲▲[Q]██▲[Q]█▲[Q]█▲[Q]█▲
      [Q]████▲▲▲"
8010 PRINT "[S][Q][Q][Q][Q][Q][Q][Q][Q][Q][Q][Q][Q][Q][Q]████
      █████[R][E]▲▲▲▲[Q]█▲[Q]█▲[Q]█▲[Q]█
      █[E]▲▲▲[Q]███▲▲▲"
8020 POKE V + 21,0
8028 FOR I = 1 TO 500 : NEXT
8030 RETURN
9000 DATA 0,0,96,0,0,224,0,1,224,0,2,224,0,2,96,0,
      4,32,0,8,0,0,8,0,0,28,0,0,62
9010 DATA 0,0,127,0,0,255,128,1,255,192
9015 DATA 1,255,192,1,255,192,1,255,192,0,255,128,0
9020 DATA 127,0,0,62,0,0,28,0,0,0,0,99
9030 DATA 0,1,128,0,2,0,0,4,0,0,4,0,0,63,0,0,255,
      128,7,255,192,15,255,224,31,255
```

```
9040 DATA 240,63,255,248,63,255,248,63,255,248,63,
       255,248,31,255,240,31,255,240
9050 DATA 31,255,192,15,255,192,7,255,128,3,255,0,1,
       255,0,0,60,0,99
9060 DATA 0,0,0,0,24,0,0,36,0,0,36,0,0,24,0,0,60,0,
       0,127,0,0,255,128,0,255,128
9070 DATA 0,255,128,1,255,192,1,255,192,3,255,224,3,
       255,224,7,255,240,7,255,240
9080 DATA 7,255,240,15,255,248,127,255,255,63,
       255,254,0,24,0
```

CHEXSUM

```
1=747
2=1705
3=3987
4=2108
5=4411
10=1171
20=946
30=1930
31=1898
32=1888
33=2435
34=2435
35=2435
36=656
37=1988
38=131
39=4986
42=4097
43=2042
44=3718
45=3005
46=1096
100=1659
110=5948
115=420
120=1354
121=1113
122=355
125=656
130=3960
131=1765
140=131
141=1609
142=2050
160=2495
161=2636
162=2498
163=2911
164=2556
165=2695
166=2915
170=1780
171=1385
172=1743
173=1745
174=1749
175=1456
178=4440
179=4386
180=2003
190=616
200=2241
210=1173
220=1667
230=1263
240=1120
250=129
5000=1624
5010=129
8000=4347
8010=4570
8020=616
8028=964
8030=143
9000=4291
9010=1762
9015=2402
9020=1632
9030=4497
9040=4467
9050=3291
9060=4274
9070=4437
9080=2908

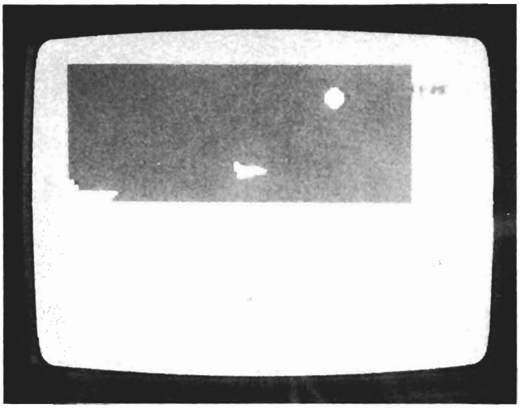
TOTAL= 165341
```



SPRITE

BELL

Sea Harrier



This is an exciting up-to-date warfare game!

You are the pilot of a carrier-based aircraft and your mission is to destroy an invading submarine using depth charges.

The enemy submarine must be destroyed in one minute, by dropping a depth charge on him, or he will escape to attack your country. When dropping your charges you must be careful not to detonate a nuclear mine placed by your own forces! The submarine will change depth and speed trying to avoid your assault.

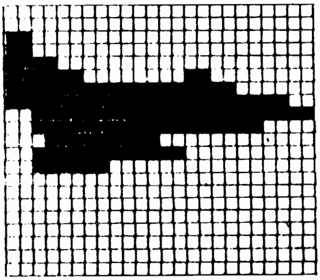
On the right is you fuel gauge, if it reaches the top you have run out of fuel and will crash. To prevent this you can return to your carrier to replenish your fuel and launch another attack on the sub. Be careful to land gently and not crash.

Your controls are:

'Z' to move up
'C' to move down
'< ' to go left
' >' to go right
'space bar' to drop depth charge

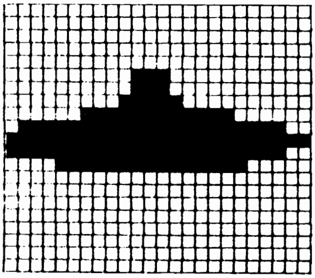
PROGRAM STRUCTURE	LINES
INITIALIZATION	1 — 10
DECREMENT FUEL GAUGE	11
MOVE SUBMARINE	12 — 14
MOVE HARRIER — INPUT KEY	20
FIRE CHARGE	25
IF FIRED MOVE THE CHARGE	26
MOVE UP	30
MOVE LEFT	40
MOVE RIGHT	50
MOVE DOWN, CHECK IF HIT SEA	56
MOVE DOWN, CHECK IF HIT SHIP	57
CHECK TIME	59
MOVE CHARGE	100
HIT SUBMARINE?	125
HIT MINE?	126 — 130
DESTROYED SUBMARINE	200 — 260
HIT NUCLEAR MINE	300 — 360
TIME UP?	400 — 450
CRASHED INTO SHIP?	500 — 550
CRASHED INTO SEA?	5000 — 5040
START AGAIN?	5050 — 5090

VARIABLES
V = VIDEO CHIP
CO = SOUND CHIP
X, Y = HORIZONTAL, VERTICAL FOR SPRITE #0 (HARRIER)
X1, Y1 = HORIZONTAL, VERTICAL FOR SPRITE#1 (SUB)
X2, Y2 = HORIZONTAL, VERTICAL FOR SPRITE #3 (CHARGE)
TI$ = TIME LEFT



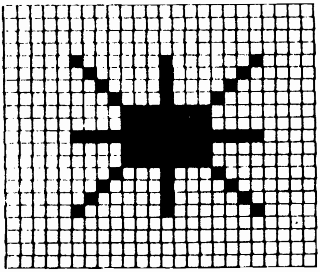
SPRITE

HARRIER
(PLAYER)



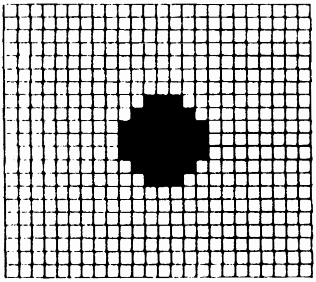
SPRITE

SUBMARINE



SPRITE

MINE



SPRITE

BOMB
(PLAYER)

SEA HARRIER

```
0 POKE 55,255 : POKE 56,47 : PRINT "▣" : V = 53248
    : CO = 54272 : POKE 53281,0 : POKE 53280,2
    : F = 10
1 FOR I = 55816 TO 56296 : POKE I,6
    : POKE I - CO,160 : NEXT : G = 0
2 POKE V + 21,63 : POKE 2040,192 : POKE 2041,193
    : POKE 2042,194 : POKE 2043,195 : POKE V + 44,7
3 POKE 2044,195 : POKE 2045,196 : POKE V + 39,1
    : POKE V + 40,14 : POKE V + 41,0
    : POKE V + 42,21
4 POKE V + 43,7 : POKE V + 29,18 : POKE V + 23,16
    : FOR I = 0 TO 318 : READ Q : POKE 12288 + I,Q
    : NEXT : X = 26
5 Y = 133 : X1 = 29 : Y1 = 160
    + INT ( RND ( 1 ) * 60 ) : POKE V,X
    : POKE V + 1,Y : D = 1 : POKE V + 2,X1
    : POKE V + 3,Y1
6 FOR J = 0 TO 24 : FOR I = 55330 TO 55335
    : POKE I + J * 40,2 : POKE I + J * 40 - CO,160
    : NEXT I,J
7 POKE V + 5,160 + INT ( RND ( 1 ) * 60 )
    : POKE V + 4,70 + INT ( RND ( 1 ) * 180 )
8 PRINT "▣▣▣▣▣▣▣▣▣▣▣▣▣▣▣▣▬▣▮▮▣▲
    ▲▲▲▬◩▣" : POKE V + 8,210 : POKE V + 9,55
    : GOSUB 80
9 POKE 54296,1 : POKE 54278,240 : POKE 54276,129
    : TI$ = "000000"
10 PRINT "▣▣▣▣▮▮▮▮▮▮▮▮▮▮▮▮▣▣TIME"
11 PRINT "▣▣▣▣▮▮▮▮▣"; RIGHT$ ( TI$,2 )
    : POKE 55652 + INT ( F ) * 40,4
12 X1 = X1 + D * 2 : IF X1 > 240
    THEN D = INT ( RND ( 1 ) * 3 - 3 )
    : Y1 = 160 + INT ( RND ( 1 ) * 60 )
13 IF X1 < 28 THEN D = INT ( RND ( 1 ) * 3 + 1 )
    : Y1 = 160 + INT ( RND ( 1 ) * 60 )
14 POKE V + 2,X1 : POKE V + 3,Y1
20 PE = PEEK ( 197 )
25 IF PE = 60 AND X > 60 AND X2 = 0 THEN X2 = X
    : Y2 = Y + 8 : G1 = 0 : F = F - .1
    : POKE 53273,200
26 IF X2 > 0 THEN GOSUB 100
27 IF F < 1 THEN 53
28 IF PE < > 12 AND PE < > 20 AND PE < > 47
    AND PE < > 44 THEN POKE 54272,0 : POKE 54273,0
    : GOTO 56
29 POKE 54273,108 : POKE 54272,223
30 IF PE = 12 AND Y > 50 THEN Y = Y - 2
40 IF PE = 47 AND X > 28 THEN X = X - 4
50 IF PE = 44 AND X < 230 THEN X = X + 6
51 F = F - .06 : IF PE = 20 THEN F = F - .1 : G = G + 2
    : GOTO 56
53 IF Y > 140 THEN 5000
54 IF F < 1 THEN Y = Y + G : GOTO 56
55 G = 0 : GOTO 58
56 Y = Y + G : G = G + .1 : IF Y > 140 THEN 5000
```

```
 57 IF X > 25 AND X < 59 AND Y > 133 AND G > 1.3
      THEN 500
 58 IF X > 25 AND X < 59 AND Y > 133 THEN G = 0
      : F = 10 : Y = 133 : GOSUB 80
 59 POKE V,X : POKE V + 1,Y : IF TI$ = "000100"
      THEN 400
 60 GOTO 11
 80 PRINT "{CLR}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{WHT}{RVS}{SPACE}{DOWN}{LEFT}{SPACE}
      {DOWN}{LEFT}{SPACE}{DOWN}{LEFT}{SPACE}{DOWN}{LEFT}{SPACE}{DOWN}{LEFT}{SPACE}{DOWN}{LEFT}{SPACE}{DOWN}{LEFT}{SPACE}{DOWN}
      {LEFT}{SPACE}{DOWN}{LEFT}{SPACE}{DOWN}{LEFT}{SPACE}"
 85 RETURN
100 X2 = X2 + .5 : IF Y2 > 140 THEN Y2 = Y2 + G1 / 2
102 IF Y2 < 141 THEN Y2 = Y2 + G1
105 IF G1 = 0 THEN POKE 54272,0 : POKE 54273,40
106 G1 = G1 + .3
110 POKE V + 7,Y2
112 POKE V + 6,X2
120 IF Y2 > 230 THEN Y2 = 0 : X2 = 0 : POKE V + 7,0
      : G1 = 0
125 IF PEEK ( V + 30 ) = 10 THEN 200
126 IF PEEK ( V + 30 ) = 12 THEN 300
130 RETURN
200 POKE V + 2,0 : POKE V + 3,0 : POKE V + 10,X1
      : POKE V + 11,Y1 : POKE 54296,15
210 FOR I = 70 TO 10 STEP - .3 : POKE 54273,I : NEXT
      : POKE 54273,0 : POKE 54272,0
220 FOR I = 1 TO 11 : POKE V + I,0 : NEXT
230 POKE V + 10,0 : POKE V + 11,0
      : PRINT "{HOME}{DOWN}{DOWN}{WHT}YOU'VE DESTROYED THE
        SUBMARINE!!!!!"
240 PRINT "{DOWN}{DOWN}{RED}WELL DONE!"
250 PRINT "{DOWN}{YEL}YOUR TIME WAS "; RIGHT$ ( TI$,2 )
      ;" SECONDS"
260 GOTO 5050
300 FOR I = 1 TO 11 : POKE V + I,0 : NEXT
      : POKE V + 29,32 : POKE V + 23,32
310 POKE V + 2,0 : POKE V + 3,0 : POKE V + 10,X2
      : POKE V + 11,Y2 : POKE 54296,15
      : POKE 53281,8
320 FOR I = 90 TO 0 STEP - .2 : POKE 54273,I
      : POKE 53280,I / 10 : NEXT : POKE 54273,0
      : POKE 54272,0
325 POKE 53280,0 : POKE 53281,0 : POKE V + 29,0
      : POKE V + 23,0
330 POKE V + 10,0 : POKE V + 11,0
      : PRINT "{HOME}{DOWN}{DOWN}{WHT}{RVS}YOU'VE HIT THE
        NUCLEAR MINE!!!!!!!!!!!!"
340 PRINT "{DOWN}{DOWN}{RED}IDIOT!!!!!"
350 PRINT "{DOWN}{YEL}YOU LASTED FOR "; RIGHT$ ( TI$,2 )
      ;" SECONDS."
360 GOTO 5050
400 POKE 54296,15 : POKE 54276,0 : POKE 54276,33
410 FOR I = 255 TO 0 STEP - 2 : POKE 54273,I : NEXT
      : POKE 54273,0 : POKE 54272,0
420 FOR I = 0 TO 11 : POKE V + I,0 : NEXT
      : POKE 53280,0 : POKE 53281,0
```

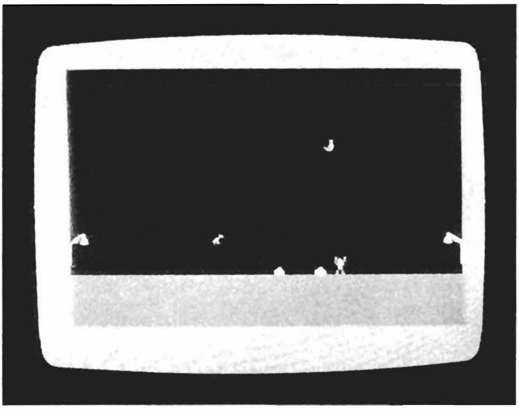
```
 430 PRINT "■ ■ ■ ■YOU'VE ▲ RUN ▲ OUT ▲ OF ▲ TIME!"
 435 PRINT "■ ■ ■ ■THE ▲ SUB ▲ WILL ▲ GO ▲ ON ▲ TO
       ▲ DESTROY ▲ YOUR ▲ ▲ ▲ ▲ ▲ ▲ COUNTRY!!!!!!"
 440 PRINT "■ ■ ■ ■YOU ▲ BUNGLING ▲ FOOL!!!!!!"
 450 GOTO 5050
 500 POKE V,0 : POKE V + 1,0 : POKE V + 10,X
       : POKE V + 11,Y : POKE 54296,15
 510 FOR I = 70 TO 20 STEP - .2 : POKE 54273,I
       : NEXT : POKE 54273,0 : POKE 54272,0
 520 FOR I = 1 TO 11 : POKE V + I,0 : NEXT
 530 POKE V + 10,0 : POKE V + 11,0 : PRINT "■ ■ ■ ■YOU'VE
       ▲ CRASHED INTO ▲ YOUR ▲ SHIP!!!!!"
 540 PRINT "■ ■TOUGH ▲ LUCK!"
 550 GOTO 5050
5000 POKE V,0 : POKE V + 1,0 : POKE V + 10,X
       : POKE V + 11,Y : POKE 54296,15
5010 FOR I = 80 TO 30 STEP - .2 : POKE 54273,I
       : NEXT : POKE 54273,0 : POKE 54272,0
5020 FOR I = 1 TO 11 : POKE V + I,0 : NEXT
5030 POKE V + 10,0 : POKE V + 11,0 : PRINT "■ ■ ■ ■YOU'VE
       ▲ CRASHED INTO ▲ THE ▲ SEA!!!!!"
5040 PRINT "■ ■ ■HARD ▲ LINES."
5050 PRINT "■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ DO ▲ YOU
       ▲ WANT ▲ ANOTHER ▲ GO?"
5060 GET A$ : IF A$ = "Y" THEN RUN
5070 IF A$ < > "N" THEN 5060
5080 PRINT "■ ■ ■BYE ▲ BYE."
5090 END
9000 DATA 0,0,0,0,0,0,192,0,0,192,0,0,240,0,0,
       252,3,0,255,255,192,255,255,252
9010 DATA 63,255,255,63,255,240,31,240,0,63,252,0,
       63,0,0,0,0,0,0,0,0,0,0,0,0,0
9020 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,99
9030 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,56,0,0,56,0,0,124,
       0,3, 255,192,127,255
9040 DATA 252,255,255,255,255,255,252,15,255,224,0,
       0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
9050 DATA 0,0,0,0,0,0,0,0,99
9060 DATA 0,0,0,0,0,0,0,,0,0,0,0,4,8,16,2,8,32,1,8,
       64,0,136,128,0,127,0,0,127,0
9070 DATA 7,255,240,0,127,0,0,127,0,0,136,128,1,8,
       64,2,8,32,4,8
9080 DATA 16,0,0,0,0,0,0,0,0,0,0,0,0,99
9090 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,ϕ,0,
       0,28,0,0,62,0,0,127,0,0
9100 DATA 127,0,0,127,0,0,62,0,0,28,0,0,0,0,0,0,0,0,
       0,0,0,0,0,0,0,0,0,0,0,0,0,0
9110 DATA 99,128,0,32,18,1,8,233,0,4,99,63,1,48,
       84,32,19,8,1,89,109,22
9120 DATA 8,12,38,0,52,149,128,243,211,99,63,0,128,
       72,1,19,8,0,89,2,22
9130 DATA 99,0,2,18,76,149,2,64,1,99,4,7,91,16,0,
       19,8,54,1,109,22
```

CHEXSUM

```
0=4547
1=3112
2=4167
3=4349
4=5113
5=6075
6=5016
7=3662
8=3639
9=2951
10=1693
11=2818
12=4928
13=3797
14=1404
20=796
25=5010
26=996
27=718
28=5265
29=1376
30=1943
40=1940
50=1997
51=3427
53=971
54=1790
55=846
56=2312
57=2713
58=3539
59=2456
60=473
80=5320
85=143
100=2708
102=1593
105=1881
106=705
110=658
112=656
120=2767
125=1456
126=1456
130=143
200=3543
210=3335
220=1586
230=5195
240=1024
250=3231
260=583
300=3144
310=4208
320=4282
325=2610
330=5203
340=931
350=3353
360=583
400=1906
410=3311
420=2804
430=2293
435=4916
440=2101
450=583
500=3128
510=3334
520=1586
530=4981
540=1099
550=583
5000=3128
5010=3339
5020=1586
5030=4708
5040=1138
5050=2839
5060=1263
5070=1188
5080=1002
5090=129
9000=4154
9010=4267
9020=1658
9030=4363
9040=4505
9050=1073
9060=4352
9070=3090
9080=1729
9090=3883
9100=4333
9110=3547
9120=3500
9130=3180

TOTAL= 258720
```

Cats & Dogs



It's raining cats and dogs! You have to catch suicidal cats and dogs but if you don't they die. For every third miss, a cross will appear, but you are only allowed three crosses.

Controls are: 'Z' to move your animal lover to the left and 'M' to move him/her right.

PROGRAM STRUCTURE	LINES
INITIALIZATION	0 — 9
CHECK ANIMALS AND MOVEMENT USING KEYS PRESSED	11 — 14
CAUGHT ANYTHING? YES: SCORE	15
NOISE	20 — 22
MOVE MAN	25 — 30
NOISE	60
IF THIS ANIMAL NOT ON SCREEN	65
PUT ONE ON (12% CHANCE)	66 — 69
ELSE MOVE HIM DOWN	70 — 85
IF HIT GROUND	86
PICTURE	90
SOUND	91 — 92
IF 3 HIT: TUNE AND CROSS	93
ELSE IF CAUGHT	94
NOISE	95 — 97
SCORE AND PICTURE	98 — 100
NEXT ANIMAL	105 — 110
DO ALL AGAIN	120
TUNE AND CROSS FOR HIT ANIMALS	1000 — 1040
IF 3 DEAD	1050
TUNE AND MESSAGES	2000 — 2110
DATA	8000 — 9045

VARIABLES
LL = LIVES LOST
P = YOUR POSITION
L = ARRAY POSITION OF ANIMALS
SP = ARRAY WHETHER CAT OR DOG

CATS AND DOGS

```
 0 L = 1 : POKE 55,255 : POKE 56,27 : V = 53248
     : FOR I = 1 TO 54 : READ R : NEXT
 1 PRINT "{CLR}" : POKE V + 32,3 : POKE V + 33,1
     : FOR I = 12544 TO 12551 : POKE I,0 : NEXT
     : POKE V + 24,28
 2 POKE 54296,10 : POKE 54276,0 : POKE 54278,240
     : FOR I = 12288 TO 12407 : READ R : POKE I,R
     : NEXT
 3 POKE 54290,0 : POKE 54292,240
 4 P = 11 : DIM L ( 3 ) ,SP ( 3 ) : FOR I = 1 TO 3
     : L ( I ) = 0 : SP ( I ) = 0 : NEXT
 5 S = 0 : SC = 1024 : CO = 54272 : CS = 55296
 6 PRINT
     "{HOME}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{RIGHT}HI{DOWN}
     ■■ ■■ ■■ ■■J{RIGHT}KM{DOWN} ■■ ■■NM{DOWN} ■■ ■■N"
     : FOR I = 1824 TO 2023
 7 POKE I + CO,2 : POKE I,4 : NEXT : POKE 198,0
 8 POKE V + 33,0 : POKE V + 32, RND ( 1 ) * 16
     : GET A$ : IF A$ = "" THEN 8
 9 POKE V + 32,6
10 FOR I = 1 TO 3
12 P1 = P : M = PEEK ( 197 )
     : IF M < > 12 AND M < > 36 THEN 50
13 P1 = P1 + ( M = 36 ) * ( P1 < 38 ) - ( M = 12 )
     * ( P1 > 1 ) : IF P1 = P THEN 50
14 PE = PEEK ( 1744 + P1 )
     : IF PE > 5 AND PE < 32 THEN 50
15 PE = PEEK ( 1784 + P1 )
     : IF PE < > 32 THEN DS = S / 2
     : IF S < 60 THEN S = S * 2 : GOTO 60
20 POKE 54277,0 : POKE 54286,240 : POKE 54276,0
     : POKE 54273,40 : POKE 54272,200
21 POKE 54276,129 : POKE 54290,0
22 POKE 54291,0 : POKE 54292,240 : POKE 54287,57
     : POKE 54286,172 : POKE 54290,17
25 POKE 1744 + P,32 : POKE 1784 + P,32
     : POKE 1704 + P,32
30 POKE 1744 + P1 + CO,3 : POKE 1744 + P1,0
     : POKE 1784 + P1 + CO,3 : POKE 1784 + P1,1
50 P = P1
60 POKE 54273,0 : POKE 54272,0 : POKE 54287,0
     : POKE 54286,0
65 IF L ( I ) < > 0 THEN 70
66 IF RND ( 1 ) > .12 THEN 110
67 L ( I ) = 1146 + INT ( RND ( 1 ) * 36 )
     : POKE 54277,0 : POKE 54286,240 : POKE 54276,0
     : POKE 54276,129
68 POKE 54273,72 : FOR J = 254 TO 235 STEP - .5
     : POKE 54272,J : NEXT : POKE 54272,0
     : POKE 54273,0
69 SP ( I ) = INT ( RND ( 1 ) * 2 )
70 L1 = L ( I )
80 L1 = L1 + 40
85 PE = PEEK ( L1 + 40 )
```

```
  86 IF PE < > 4 AND PE < > 5 THEN 95
  90 POKE L1 + 40,4 : POKE L1 + CO,4 + SP ( I )
       : POKE L1,5 : POKE L ( I ) ,12
  91 POKE 54277,0 : POKE 54286,240 : POKE 54276,0
       : POKE 54276,129 : POKE 54273,34
  92 FOR J = 150 TO 135 STEP - 1 : POKE 54272,J
       : NEXT : POKE 54272,0 : LL = LL + 1
  93 POKE L ( I ) ,32 : L ( I ) = 0
       : IF LL = 3 THEN 1000
  94 GOTO 110
  95 IF PE < > 0 AND PEEK ( L1 ) < > 0 THEN 100
  96 POKE L ( I ) ,7 : POKE 54296,10 : POKE 54291,0
       : POKE 54290,0 : POKE 54292,240
       : POKE 54290,33
  97 POKE 54287,64 : POKE 54286,188 : FOR J = 1 TO 50
       : NEXT : POKE 54287,0 : POKE 54286,0
  98 POKE L ( I ) ,32 : L ( I ) = 0
       : S = ( S + ( LL + 10 ) ) + ( 2 * L )
       : GOTO 110
 100 POKE L ( I ) ,32 : POKE L1 + CO,4 + SP ( I )
       : POKE L1,2 + SP ( I )
 105 L ( I ) = L1
 110 NEXT
 120 GOTO 10
1000 L = L + 1 : POKE 54287,0 : POKE 54276,0
       : POKE 54283,0 : POKE 54272,0 : RESTORE
       : POKE 54296,15
1010 PRINT "[CLR]"; TAB( 18 + ( L * 4 ) ) ;"F"
       : POKE 54276,0 : POKE 54290,0 : POKE 54292,0
       : POKE 54278,0
1015 POKE V + 32,0 : POKE 54291,9 : POKE 54277,96
1020 FOR Z = 1 TO 18 : READ A,B,LE : B1 = B * 2 + 1
       : A1 = A * 2 : IF B1 > 255 THEN A1 = A1 + 1
       : B1 = B1 - 257
1025 POKE 54287,A : POKE 54286,B : POKE 54273,A1
       : POKE 54272,B1 : POKE 54290,17
1030 POKE 54276,33 : FOR K = 1 TO LE : NEXT
       : POKE 54273,0 : POKE 54272,0 : POKE 54287,0
1035 POKE 54286,0 : POKE 54290,0 : POKE 54276,0
       : NEXT Z
1040 POKE V + 32,6 : LL = 0 : POKE 54278,240
       : POKE 54292,240 : POKE 54277,0 : POKE 54291,0
1050 POKE 54296,10 : IF L < 4 THEN 94
2000 POKE V + 32,0 : POKE V + 24,20 : PRINT "[CLR]"
2010 PRINT "[ ]HARD LUCK"
2020 PRINT "[ ]YOU SCORED"S"POINTS"
2030 PRINT "[ ] [ ]ANOTHER GAME?"
2040 PRINT "[ ]OR WAS THE LAST ONE TOO TOUGH"
2070 GET A$ : IF A$ = "Y" THEN RUN
2080 IF A$ < > "N" THEN 2070
2090 PRINT "[ ]BYE BYE THEN."
2100 PRINT "[ ]YOUR COMPUTER IS";
2110 END
8000 DATA 34,75,400,0,0,40,34,75,300,0,0,40,34,75,
       100,34,75,400,0,0,40
8010 DATA 40,200,300,38,126,100,0,0,40,38,126,300,34,
```

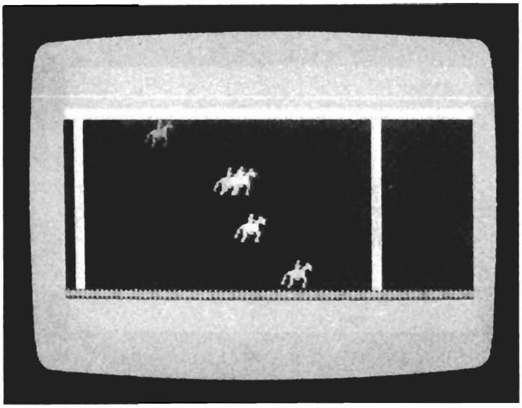
```
      75,100,0,0,40,34,75,300
8020 DATA 0,0,40,34,75,100,0,0,40,34,75,500
9000 DATA 0,0,129,129,185,186,146,124,56,56,56,40,68,
      68,68,198
9010 DATA 4,7,4,56,112,208,72,72,14,14,4,14,31,142,
      110,60
9015 DATA 255,255,255,255,255,255,255,255
9020 DATA 0,0,28,60,254,255,255,255,24,24,255,255,24,
      24,24,24
9025 DATA 0,73,42,0,99,0,42,73
9030 DATA 0,0,3,13,25,62,96,192,56,232,28,28,62,62,
      127,8
9040 DATA 28,22,59,56,124,124,254,16,0,0,192,176,152,
      124,6,3
9050 DATA 0,0,0,0,146,68,1,82,128,128,128,128,128,128,
      128,128,1,1,1,1,1,1,1,1
```

CHEXSUM

```
0=3393
1=4571
2=4360
3=1279
4=3520
5=2356
6=4157
7=1746
8=3201
9=619
10=656
12=3038
13=4202
14=2754
15=4912
20=3315
21=1269
22=3397
25=2595
30=4256
50=395
60=2393
65=1101
66=1169
67=4537
68=4258
69=1350
70=554
80=721
85=1039
86=1786
90=3206
91=3328
92=3684
93=2156
94=526
95=2088
96=3837
97=3711
98=3705
100=3010
105=555
110=131
120=472
1000=3979
1010=3965
1015=1913
1020=5631
1025=3336
1030=3547
1035=2089
1040=3836
1050=1402
2000=1852
2010=933
2020=1773
2030=1267
2040=2706
2070=1263
2080=1184
2090=1204
2100=1745
2110=129
8000=3465
8010=4034
8020=1959
9000=3015
9010=2737
9015=1886
9020=2945
9025=1214
9030=2710
9040=2917
9050=4197

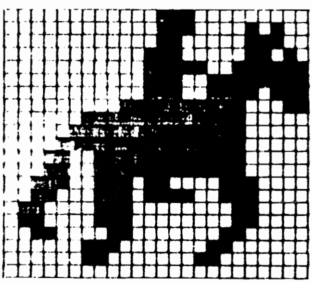
TOTAL= 184141
```

Horse Racing

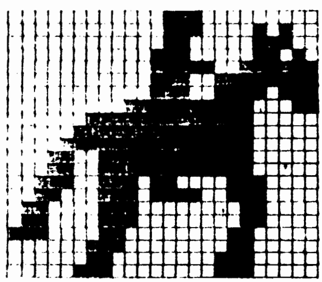


Bet on the horses without losing anything. The game begins with the names of the four horses displayed against their odds. You must choose a horse and place your bet.

Program contains full instructions.



SPRITE
HORSE
SHAPE
No. 2



SPRITE
HORSE
SHAPE
No. 1

PROGRAM STRUCTURE	LINES
INITIALIZATION	1 — 9
DISPLAY PLAYER'S FINANCE	20 — 21, 500
GENERATE ODDS	11
DISPLAY ODDS	25 — 36
PROMPTING FOR BETS	20
SETUP RACE DISPLAY	100 — 120
RUN THE RACE	130 — 140
DISPLAY RESULTS	205 — 5040
DATA FOR 'SPRITE' CHARACTERS	9000 — 9050

VARIABLES
M = PLAYER'S MONEY
OD() = ODDS OF EACH HORSE
B = PLAYER'S BET
H = HORSE CHOSEN
H() = POSITION OF EACH HORSE
Z = DELAY LOOP VARIABLE
HN = WINNING HORSE

HORSE RACING

```
0 REM * A DAY AT THE RACES *
1 M = 100 : CO = 54272 : POKE 53280,0
   : POKE 53281,0
2 FOR N = 0 TO 126 : READ Q : POKE 832 + N,Q
   : NEXT N
3 V = 53248 : POKE V + 21,15 : POKE 2040,13
   : POKE 2041,13 : POKE 2042,13 : POKE 2043,13
4 POKE V + 39,2 : POKE V + 40,3 : POKE V + 41,4
   : POKE V + 42,5
5 POKE 54296,15 : POKE 54277,64 : POKE 54278,128
   : POKE 54276,129 : POKE 54273,0
9 POKE 54272,0
10 PRINT "                A DAY AT
   THE RACES"
11 FOR I = 1 TO 4 : OD ( I ) = INT ( RND ( 1 )
   * 10 + 1 ) : NEXT
20 PRINT "   YOU HAVE £";M;"LEFT."
21 IF M = 0 THEN 500
25 PRINT "   HORSE                 ODDS"
26 PRINT "
   "
30 PRINT "  1.  RED DEVIL       "
   ;OD ( 1 ) ;"/1"
32 PRINT "  2.  THE BLUE MAX    ";OD ( 2 )
   ;"/1"
34 PRINT "  3.  PINK LADY       "
   ;OD ( 3 ) ;"/1"
36 PRINT "  4.  GREEN GOBLIN    ";OD ( 4 )
   ;"/1"
40 PRINT "   . BET HOW MUCH"
50 INPUT B : IF B > M THEN PRINT "
   " : GOTO 40
60 PRINT " ON WHICH HORSE (1,2,3,4)"
70 INPUT H : IF H < 1 OR H > 4 THEN PRINT "
   " : GOTO 60
80 PRINT "                   PRESS
   ANY KEY                ";
90 GET A$ : IF A$ = "" THEN 90
100 PRINT "" : POKE 53281,5 : POKE 53280,8
   : FOR I = 55496 TO 56135 : POKE I,0
   : POKE I - CO,160
101 NEXT
102 FOR I = 55456 TO 55495 : POKE I,1
   : POKE I - CO,35 : NEXT
103 FOR I = 56136 TO 56175 : POKE I,0
   : POKE I - CO,163 : NEXT
105 PRINT "
    F I N
   I S H       "
106 PRINT "
        S T A R T
   "
110 H ( 1 ) = 20 : H ( 2 ) = 20 : H ( 3 ) = 20
   : H ( 4 ) = 20 : LE = 1
```

```
120 FOR I = 0 TO 3 : POKE V + I * 2,H ( I + 1 )
      : POKE V + I * 2,90 + ( I * 35 ) : NEXT
125 IF LE = 1 THEN LE = 0 : GOTO 130
126 IF LE = 0 THEN LE = 1
130 FOR I = 0 TO 3
131 POKE 2040 + I,13 + LE : IF LE = 0
      THEN POKE 54273,2 : POKE 54272,37
132 IF RND ( 1 ) < .45 OR ( OD ( I + 1 ) < 5
      AND RND ( 1 ) < .4 ) THEN H ( I + 1 )
      = H ( I + 1 ) + INT ( RND ( 1 ) * 5 )
      : GOTO 135
133 H ( I + 1 ) = H ( I + 1 ) + 1
135 POKE V + I * 2,H ( I + 1 ) : POKE V + 1 + I
      * 2,90 + ( I * 35 ) : IF LE = 0
      THEN POKE 54273,0 : POKE 54272,0
136 IF H ( I + 1 ) > 240 THEN 200
140 NEXT : GOTO 125
200 FOR Z = 1 TO 500 : NEXT
205 POKE 53281,0 : POKE 53280,0 : PRINT "■ ◣"
      : FOR Z = 0 TO 3 : POKE V + Z * 2,0 : NEXT
      : HN = I + 1
206 GOSUB 5000
210 PRINT " ▲ ■WINS."
220 PRINT "■YOU ▲ BET ▲ ON ▲ "; : HN = H : GOSUB 5000
230 IF I + 1 < > H THEN 300
240 PRINT " ▲ TOO!"
245 M = M + OD ( I + 1 ) * B
250 PRINT "■SO ▲ YOU'VE ▲ WON ▲ £";OD ( I + 1 ) * B
260 GOTO 30
300 PRINT : PRINT "■YOU'VE ▲ LOST ▲ £";B
310 PRINT "■ ■BAD ▲ LUCK!"
315 M = M - B
320 IF M < 1 THEN FOR I = 1 TO 3000 : NEXT : GOTO 10
350 PRINT "DO ▲ YOU ▲ WANT ▲ TO ▲ PLAY ▲ AGAIN?"
360 GET A$ : IF A$ = "Y" THEN 10
370 IF A$ < > "N" THEN 360
380 PRINT "■ ■ ■BYE ▲ BYE!"
390 END
500 PRINT "■YOU'RE ▲ BROKE!"
505 M = 100
510 GOTO 350
5000 IF HN = 1 THEN PRINT "■RED ▲ DEVIL";
5010 IF HN = 2 THEN PRINT "■THE ▲ BLUE ▲ MAX";
5020 IF HN = 3 THEN PRINT "■THE ▲ PINK ▲ LADY";
5030 IF HN = 4 THEN PRINT "■GREEN ▲ GOBLIN";
5040 RETURN
9000 DATA 0,14,0,0,14,20,0,12,28,0,28,26,0,31,
      63,0,28,255,0,28,247,0,127,227,3
9010 DATA 255,224,7,255,224,15,255,224,25,255,224,
      25,255,224,57,216,112,49,222
9020 DATA 112,97,192,48,97,192,48,225,128,96,3,
      128,96,3,0,224,7,0,224,99
9030 DATA 0,14,0,0,14,20,0,12,28,0,28,26,0,31,63,0,
      28,255,0,28,247,0,127,227,3
9040 DATA 255,224,7,255,224,15,255,224,25,255,224,25,
      255,224,57,216,112,49,222
```

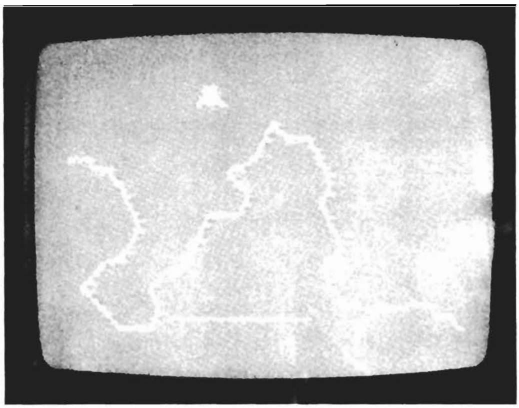
```
9050 DATA 112,97,192,48,97,192,48,48,192,96,3,
     128,192,3,1,128,0,0,0
```

CHEXSUM

```
0=0
1=2358
2=2201
3=3798
4=2787
5=3423
9=527
10=2413
11=2632
20=1824
21=782
25=1873
26=4167
30=2107
32=2309
34=2210
36=2216
40=1462
50=3006
60=1890
70=3450
80=2588
90=1150
100=4266
101=131
102=2647
103=2717
105=7151
106=5027
110=2852
120=3916
125=1704
126=1092
130=655
131=3102
132=5952
133=1349
135=5250
136=1338
140=718
200=979
205=4619
206=352
210=652
220=2077
230=1234
240=545
245=1301
250=2399
260=472
300=1643
310=906
315=590
320=2266
350=2300
360=1243
370=1125
380=852
390=129
500=1359
505=411
510=526
5000=1754
5010=1986
5020=2227
5030=2039
5040=143
9000=4274
9010=4325
9020=3597
9030=4274
9040=4325
9050=3299

TOTAL= 161253
```

Luna Landa



This is one for all you budding young astronauts. The surface of the moon is looming closer, and you must control the horizontal and vertical velocities of your module to achieve a feather-light landing.

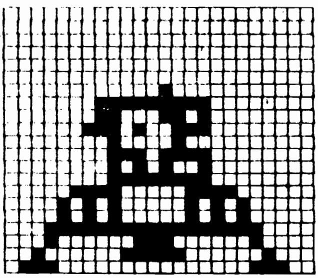
Keep an eye on your fuel gauge at the bottom of the screen, don't run out of fuel. The more difficult the landing site the more points you get for a successful mission, each perfect landing increases the difficulty of the next. Be careful not to damage your module by landing too hard.

Controls are:

- 'C' move right
- 'Z' move left
- 'F3' ¼ power thrust
- 'F5' ½ power thrust
- 'F7' full power thrust

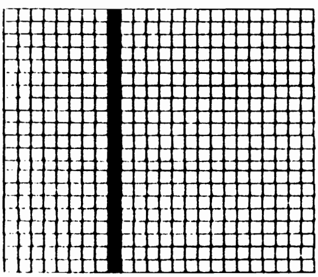
PROGRAM STRUCTURE	LINE
INITIALIZATION (SET GRAVITY, VELOCITIES (HORIZONTAL AND VERTICAL) LINE 35)	1 — 35
INITIALIZE LUNAR MODULE	44
AFFECT MODULE WITH GRAVITY	50 — 80
CONTINUE HORIZONTAL & VERTICAL MOTION INPUT KEY	85
CHANGE HORIZONTAL VELOCITY IF C OR Z	87
CHANGE VERTICAL VERLOCITY IF THRUST	88 — 92
CHECK LANDING ON PAD 1 Y? THEN 1000	100
CHECK LANDING ON PAD 2 Y? THEN 1000	105
CHECK LANDING ON PAD 3 Y? THEN 1000	110
CHECK FOR CRASH	115
IF CRASHED, EXPLODE	120 — 130
ASK FOR NEW GAME	135
YES? THEN 160	140
END	150
REDRAW LANDSCAPE, REINITIALIZE VELOCITIES, POSITIONS, ETC GO TO MAIN BODY	160 — 180
LANDED SAFELY? N? THEN 120 GIVE SCORE FOR SAFE LANDING INCREASE DIFFICULTY FOR NEXT TIME	1000 — 1010
DATA	1000 — 10080

VARIABLES
V = ADDRESS OF VIDEO CHIP
G = GRAVITY
VV = VERTICAL VELOCITY
HV = HORIZONTAL VELOCITY
NO = NO. OF SAFE LANDINGS
X1 = X POSITION
Y1 = Y POSITION
K = KEYBOARD PORT
K = KEY PRESSED
FU = FUEL



SPRITE

LUNA
MODULE



SPRITE

FUEL
MARKER

LUNA LANDA

```
  1 POKE 55,255 : POKE 56,47 : V = 53248
      : FOR I = 12544 TO 12551 : POKE I,0 : NEXT
  5 FOR I = 0 TO 103 : READ A : POKE 12 * 1024 + I,A
      : NEXT
 10 FOR I = 832 TO 894 : READ A : POKE I,A : NEXT
      : POKE 2040,13 : POKE V + 24,28
      : POKE V + 39,15 : NO = 1
 15 POKE V + 33,0 : POKE V + 32,0 : FU = 1000
      : SC = 0
 20 PRINT "♥ ▤ ▣ ▣ ▣ ▣ ▣ ▣ ▣ ▣GG ▣BC ▣D ▣H ▣ ██J ▣H
      ▣ ██J ▣ ██I ▣ ██K ▣ ██ ██E ▣ ██ ██ ██∂A ▣ ██ ██
      ██E ▣ ██ ██E ▣ ██D ▣D ▣D ▣DLLLK ▢ ██I";
 25 PRINT "▢ ██J ▢ ██H ▢ ██E ▢E ▢∂A ▢E ▢K ▢ ██I ▢K ▢ ██I
      ▢LLLK ▢ ██I ▢ ██D ▢ ██ ██D ▢ ██∂A ▢E ▢K ▢
      ██I ▢F ▣D GG ▣H ▣ ██J ▣H ▣ ██J";
 30 PRINT "▣ ██I ▣ ██K ▣ ██H ▣ ██J ▣H ▣ ██J ▣ ██E ▣ ██ ██E
      ▣ ██H ▣ ██J ▣D ▣BCLLLLEBC ▣BC ▣D";
 35 G = .040875 : VV = 0 : HV = .5 : POKE V,2
      : POKE V + 1,29 : POKE V + 16,0
      : POKE V + 21,1 : KB = 197 : X = 2
 40 FOR I = 0 TO 190 : READ A : POKE 13 * 1024 + I,A
      : NEXT I
 42 FOR I = 896 TO 958 : POKE I,0
      : IF INT ( ( I + 1 ) / 3 ) * 3 = I
      THEN POKE I,128
 43 NEXT
 44 POKE V + 3,232 : POKE V + 21, PEEK ( V + 21 )
      OR 2 : POKE 2041,14 : POKE V + 40,2
      : A = PEEK ( V + 31 )
 45 PRINT "▨ ▣ ▣ ▣ ▣ ▣ ▣ ▣ ▣ ▣ ▣ ▣ ▣ ▣ ▣ ▣ ▣ ▣ ▣ ▣
      ▣ ▣ ▣ ▣ ◘ ◘ ◘ ◘ ◘ ◘ ◘ ◘ ◘ ◘ ▩LLLLLLLLLLLLLL";
 50 VV = VV + G : POKE V + 2,FU / 10 + 96
      : IF VV > 4 THEN VV = 4
 55 A = PEEK ( V + 1 ) : A = A + VV : IF A > 230
      THEN A = 230
 57 IF A < 29 THEN A = 29
 60 POKE V + 1,A : B = PEEK ( V + 16 ) AND 1
 65 A = X + HV : IF A > 255 AND B = 0
      THEN POKE V + 16, PEEK ( V + 16 ) OR 1 : B = 1
      : A = 0
 70 IF A < 0 AND B THEN POKE V + 16, PEEK ( V + 16 )
      AND 254 : B = 0 : A = 255
 75 IF A < 0 THEN POKE V + 16, PEEK ( V + 16 ) OR 1
      : B = 1 : A = 87
 80 IF A > 87 AND B THEN POKE V + 16, PEEK ( V + 16 )
      AND 254 : B = 0 : A = 0
 85 POKE V,A : X = A : K = PEEK ( KB ) : IF FU < 1
      THEN K = 0
 87 HV = HV + ( K = 20 ) * ( HV < 4 ) * .1 -
      ( K = 12 ) * ( HV > - 4 ) * .1 : FU = FU +
      ( K = 20 ) + ( K = 12 )
 88 IF ( FU < 3 AND K = 6 ) OR ( FU < 4 AND K = 3 )
```

```
      THEN K = 0
 90 VV = VV + ( K = 5 ) * .0625 + ( K = 6 ) * .125
      + ( K = 3 ) * .25
 92 FU = FU + ( K = 5 ) + ( K = 6 ) * 3 + ( K = 3 )
      * 5
 95 Y1 = PEEK ( V + 1 ) : X1 = PEEK ( V )
      : B = PEEK ( V + 16 ) AND 1
100 IF Y1 = 228 AND X1 > = 63 AND X1 < = 67
      THEN LA = 1 : GOTO 1000
105 IF Y1 = 132 AND X1 > = 143 AND X1 < = 147
      THEN LA = 2 : GOTO 1000
110 IF Y1 = 204 AND X1 > = 7 AND X1 < = 13 AND B
      THEN LA = 3 : GOTO 1000
115 A = PEEK ( V + 31 ) AND 1 : IF A < > 1 THEN 50
120 FOR I = 208 TO 210.9 STEP .025 : POKE 2040,I
      : NEXT : FOR I = 210.5 TO 209 STEP - .05
125 POKE 2040,I : NEXT : POKE V + 21,0
130 FOR I = 1 TO 5000 : NEXT : PRINT "♥"
      : POKE V + 21,0 : POKE V + 24,20
      : PRINT "▣ ▣ ▣ ▦YOU ▲ CRASHED!"
135 PRINT "▣ ▣ ▣ ▥SCORE";SC
      : PRINT "▣ ▣ ▣ ▩ANOTHER ▲ GAME?"
140 GET A$ : IF A$ = "Y" THEN 160
145 IF A$ < > "N" THEN 140
150 END
160 FU = 1000 : SC = 0 : POKE 2040,13
      : POKE V + 24,28
165 PRINT "♥ ☰ ▣ ▣ ▣ ▣ ▣ ▣ ▣ ▣GG ▣BC ▣D ▣H ▣ ■■J ▣H
      ▣ ■■J ▣ ■■I ▣ ■■K ▣ ■■ ■■E ▣ ■■ ■■ ■■∂A ▣ ■■ ■■
      ■■E ▣ ■■ ■■E ▣ ■■D ▣D ▣D ▣DLLLK □ ■■I";
170 PRINT "□ ■■J □ ■■H □ ■■E □E □∂A □E □K □ ■■I □K □ ■■I
      □LLLK □ ■■I □ ■■D □ ■■ ■■D □ ■■∂A □E □K □
      ■■I □F ▣DGG ▣H ▣ ■■J ▣H ▣ ■■J";
175 PRINT "▣ ■■I ▣ ■■K ▣ ■■H ▣ ■■J ▣H ▣ ■■J ▣ ■■E ▣ ■■ ■■E
      ▣ ■■H ▣ ■■J ▣D ▣BCLLLLEBC ▣BC ▣D";
180 VV = 0 : HV = .5 : POKE V,2 : POKE V + 1,29
      : POKE V + 16,0 : POKE V + 21,1 : X = 2
      : GOTO 44
1000 IF VV > 5 OR ABS ( HV ) > .4 THEN 120
1005 SC = SC + ( 4 - LA ) * 50 : FOR I = 1 TO 5000
      : NEXT : NO = NO + 1
1010 POKE V,2 : POKE V + 1,29 : POKE V + 16,0
      : POKE V + 21,1 : KB = 197 : X = 2
      : HV = NO * .5 : GOTO 44
10000 DATA 0,0,0,5,10,16,96,128,1,5,8,144,96,0,0,0,
      128,64,64,80,105,6,0,0
10005 DATA 0,0,0,0,132,74,49,1,128,64,48,8,4,4,6,1,
      1,1,2,10,20,16,32,192
10010 DATA 0,0,8,20,36,66,65,129,0,0,0,0,4,42,209,
      129,128,64,64,32,16,16,8,8
10015 DATA 1,1,2,4,2,2,4,8,4,2,2,4,2,2,1,1,8,16,
      32,32,32,64,64,128
10020 DATA 0,0,0,0,0,0,0,255
10025 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
      8,0,1,255,0,1,141,0,3,167,0,0
10030 DATA 132,0,0,239,0,0,169,0,0,255,0,7,131,
```

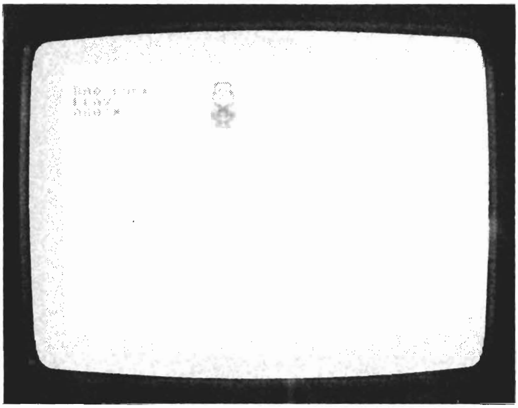
```
        192,10,130,160,10,130,160,31
10035 DATA 255,240,16,56,16,32,124,8,112,0,28
10050 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
        0,0,0,0,0,0,0,0,8,0,0,8,0,0
10055 DATA 20,0,34,0,0,20,0,0,8,0,0,8,0,0,0,0,0,0,
        0,0,0,0,0,0,0,0,0,0,99
10060 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
        0,0,0,0,0,0,8,0,0,54,0,0,193
10065 DATA 128,1,0,64,6,0,48,1,0,64,0,193,128,0,54,
        0,0,8,0,0,0,0,0,0,0,0,0,0,0
10070 DATA 0,99,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
        0,0,0,0,0,0,0,0,0,0,28,0,0
10075 DATA 227,128,15,0,120,112,0,7,15,0,120,0,227,
        128,0,28,0,0,0,0,0,0,0,0,0,0
10080 DATA 0,0,0,0,0,0
```

CHEXSUM

```
1=3492
5=2434
10=4629
15=2449
20=7077
25=9367
30=4607
35=5451
40=2530
42=3732
43=131
44=4772
45=2992
50=3406
55=2875
57=1042
60=1840
65=4349
70=3395
75=2942
80=3369
85=2696
87=6197
88=2856
90=3582
92=2820
95=3021
100=3547
105=3666
110=3743
115=2121
120=3765
125=1399
130=4842
135=2795
140=1306
145=1125
150=129
160=2430
165=7077
170=9367
175=4607
180=4562
1000=1784
1005=3464
1010=5069
10000=3608
10005=3506
10010=3997
10015=3159
10020=1004
10025=4116
10030=3833
10035=2013
10050=4096
10055=3679
10060=3998
10065=4387
10070=4121
10075=4255
10080=692

TOTAL= 215315
```

Simon

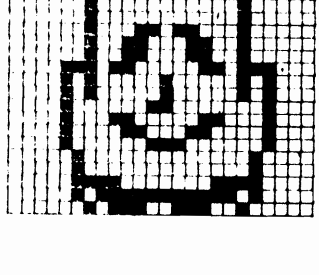


Simon plays a series of tones incrementing the number each time. You must repeat the sequence. The screen displays the numbers 1 — 6 and one of these will light up to correspond with the tones played. Each time Simon repeats a sequence he adds a note and plays the 'tune' faster. You win by repeating his tune until it is eight notes long. Use the keys 1 — 6 to reproduce the tones and lights.

Remember, this game tests your speed as well as your memory and you only have a limited amount of time to react!

SPRITE

SIMON'S HEAD



PROGRAM STRUCTURE	LINES
INITIALIZATION	0 — 3
DRAW SIMON, CAPTION: CALL 'DRAW LIGHTS'	4 — 9
CALL INTRO' DIMENSION NOTE ARRAY	10
PLAY NOTE AND ENTER INTO ARRAY: CALL 'LIGHT KEY'	20 — 25
CLEAR NOTE: CALL 'LIGHT KEY'	30 — 35
PROMPT FOR KEY INPUT	40
GET KEY (IF ANY) FROM QUEUE AT END OF DELAY	45 — 50
WRONG KEY, CALL 'END'	55
COMPARE WITH ARRAY, IF WRONG CALL 'END'	56 — 80
PRINT CONGRATULATIONS AND GET KEY TO RESTART	90 — 99
PRINT LIGHTS ONTO SCREEN	100 — 190
CHANGE COLOUR OF SELECTED LIGHT	200 — 270
TURN OFF SOUNDS AND GET KEY TO RESTART	500 — 560
PLACE CAPTIONS IN BUBBLE AND DELAY	5000 — 5140
DELAY AND CLEAR CAPTION	6000 — 6030
DATA FOR SIMON	9000 — 9050

VARIABLES
V = POINTER FOR SPRITE HANDLING
N(X) = NOTE ARRAY
I = POINTER TO LAST NOTE PLAYED (I = −1, INTRO CAPTIONS IN PROGRESS)
A = PLAYER'S INPUT

SIMON

```
  0 PRINT "▣" : POKE 53281,1 : POKE 53280,4 :
      POKE 54296,15 : POKE 54276,0 : POKE 54276,17
  1 POKE 54277,0 : POKE 54278,240 : POKE 54273,0 :
      V = 53248 : POKE V + 21,3 : POKE 2040,13
  2 POKE 2041,14 : POKE V + 39,0 : POKE V + 40,5 :
      POKE V + 28,2 : POKE V + 38,0 :
      FOR I = 0 TO 126 : READ Q
  3 POKE 832 + I,Q : NEXT : POKE V,140 :
      POKE V + 2,140 : POKE V + 1,61 : POKE V + 3,82
  4 PRINT "▣▣▲□□□□□□□□□"
  5 PRINT "□▲▲▲▲▲▲▲▲▲□□"
  6 PRINT "□▲▲▲▲▲▲▲▲▲▲▣"
  7 PRINT "□▲▲▲▲▲▲▲▲▲▲□"
  8 PRINT "▲□□□□□□□□□□"
  9 PRINT "◣▣◆▲◆▲◆▲◆▲◆▲◆▲◆▲◆▲◆
      ▲◆▲◆▲◆▲◆▲◆▲◆▲◆▲◆▲◆▲◆
      ▲◆▲" : GOSUB 100
 10 GOSUB 5000 : DIM N ( 8 )
 20 FOR I = 1 TO 8 : N ( I ) =
      INT ( RND ( 1 ) * 6 ) + 1 : FOR J = 1 TO I :
      POKE 54276,17
 25 POKE 54273,N ( J ) * 20 : PRINT "▣" :
      GOSUB 200
 30 FOR X = 1 TO 400 - I * 30 : NEXT : POKE 54273,0
 35 PRINT "◣" : GOSUB 200
 40 FOR X = 1 TO 200 - I * 20 : NEXT X,J : PRINT
      "▣▣▣▣▣▲REPEAT▣■■■■■THE▣■■■
      NOTES!"
 45 FOR J = 1 TO I : POKE 54276,0 : POKE 54276,33 :
      POKE 198,0 : FOR X = 1 TO 250 - I * 20
 50 GET A$ : IF VAL ( A$ ) > 0 AND VAL ( A$ ) < 7
      THEN 55
 51 NEXT X : GOTO 500
 55 GOSUB 6000
 56 IF VAL ( A$ ) < > N ( J ) THEN 500
 60 POKE 54273, VAL ( A$ ) * 20 : PRINT "▣" :
      GOSUB 200
 62 FOR X = 1 TO 500 - I * 50 : NEXT : POKE 54273,0
      : PRINT "◣" : GOSUB 200
 80 FOR X = 1 TO 1500 - I * 80 : NEXT J :
      FOR X = 1 TO 600 : NEXT X,I
 90 GOSUB 6000 : PRINT "▣▣▣▣▣▲YOU▣■■■
      ▲WIN!!"
 91 GOSUB 6000 : PRINT "▣▣▣▣▣▲PLAY▣■■■
      ■AGAIN?"
 92 GET A$ : IF A$ = "Y" THEN 20
 95 IF A$ < > "N" THEN GOTO 92
 98 GOSUB 6000 : PRINT "▣▣▣▣▣▲▄BYE!"
 99 END
100 PRINT "▣▣▣▣▣▣▣▣▣▣▣▣▣▣▣▣▣◩
      ▲▲▲◥▣▣▣▣▣▣▣▣▣▣▣▣▣▣◩▲▲▲◥"
110 PRINT "▣▣▣▣▲▲1▲▲▣▣▣▣▣▣▣▣▣▣
      ▣▣▲▲4▲▲"
120 PRINT "▣▣▣◥▣▲▲▲▀◩▣▣▣▣▣
```

```
        {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} ◥ {RVS} ▲ ▲ ▲ ▬ ◢"
130 PRINT "{DOWN} {DOWN} {RIGHT} {RIGHT} {RIGHT} {RVS} ◢ ▲ ▲ ▲ ◥ {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT}
        {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} ◢ ▲ ▲ ▲ ◥"
140 PRINT "{RIGHT} {RIGHT} {RIGHT} {RVS} ▲ ▲2▲ ▲ {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT}
        {RIGHT} {RIGHT} ▲ ▲5▲ ▲"
150 PRINT "{RIGHT} {RIGHT} {RIGHT} ◥ {RVS} ▲ ▲ ▲ ▬ ◢ {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT}
        {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} ◥ {RVS} ▲ ▲ ▲ ▬ ◢"
160 PRINT "{DOWN} {DOWN} {RIGHT} {RIGHT} {RIGHT} {RVS} ◢ ▲ ▲ ▲ ◥ {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT}
        {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} ◢ ▲ ▲ ▲ ◥"
170 PRINT "{RIGHT} {RIGHT} {RIGHT} {RVS} ▲ ▲3▲ ▲ {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT}
        {RIGHT} {RIGHT} {RIGHT} ▲ ▲6▲ ▲"
180 PRINT "{RIGHT} {RIGHT} {RIGHT} ◥ {RVS} ▲ ▲ ▲ ▬ ◢ {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT}
        {RIGHT} {RIGHT} {RIGHT} {RIGHT} ◥ {RVS} ▲ ▲ ▲ ▬ ◢"
190 RETURN
200 PRINT "{CLR} {DOWN} {DOWN} {DOWN} {DOWN} {DOWN} {DOWN} {DOWN} {DOWN} {DOWN} {RIGHT} {RIGHT} {RIGHT}"; :
       IF N ( J ) > 3 THEN PRINT
       "{RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT} {RIGHT}";
210 IF N ( J ) = 1 THEN PRINT "{RVS} ◢ ▲ ▲ ▲ ◥ {DOWN} ▮
        ▮ ▮ ▮ ▮ ▲ ▲1▲ ▲ {DOWN} ▮ ▮ ▮ ▮ ▮ ▬ ◥ {RVS}
        ▲ ▲ ▲ ▬ ◢"
220 IF N ( J ) = 2 THEN PRINT "{DOWN} {DOWN} {DOWN} {DOWN} {DOWN} {RVS}
        ◢ ▲ ▲ ▲ ◥ {DOWN} ▮ ▮ ▮ ▮ ▮ ▲ ▲2▲ ▲ {DOWN} ▮ ▮ ▮ ▮
        ▮ ▬ ◥ {RVS} ▲ ▲ ▲ ▬ ◢"
230 IF N ( J ) = 3 THEN PRINT "{DOWN} {DOWN} {DOWN} {DOWN} {DOWN} {DOWN} {DOWN} {DOWN}
        {DOWN} {DOWN} {RVS} ◢ ▲ ▲ ▲ ◥ {DOWN} ▮ ▮ ▮ ▮ ▮ ▲ ▲3▲ ▲ {DOWN}
        ▮ ▮ ▮ ▮ ▮ ▬ ◥ {RVS} ▲ ▲ ▲ ▬ ◢"
240 IF N ( J ) = 4 THEN PRINT "{RVS} ◢ ▲ ▲ ▲ ◥ {DOWN} ▮
        ▮ ▮ ▮ ▮ ▲ ▲4▲ ▲ {DOWN} ▮ ▮ ▮ ▮ ▮ ▬ ◥ {RVS}
        ▲ ▲ ▲ ▬ ◢"
250 IF N ( J ) = 5 THEN PRINT "{DOWN} {DOWN} {DOWN} {DOWN} {DOWN} {RVS}
        ◢ ▲ ▲ ▲ ◥ {DOWN} ▮ ▮ ▮ ▮ ▮ ▲ ▲5▲ ▲ {DOWN} ▮ ▮ ▮ ▮
        ▮ ▬ ◥ {RVS} ▲ ▲ ▲ ▬ ◢"
260 IF N ( J ) = 6 THEN PRINT "{DOWN} {DOWN} {DOWN} {DOWN} {DOWN} {DOWN} {DOWN} {DOWN}
        {DOWN} {DOWN} {RVS} ◢ ▲ ▲ ▲ ◥ {DOWN} ▮ ▮ ▮ ▮ ▮ ▲ ▲6▲ ▲ {DOWN} ▮ ▮ ▮
        ▮ ▮ ▬ ◥ {RVS} ▲ ▲ ▲ ▬ ◢"
270 RETURN
500 POKE 54276,0 : POKE 54276,33
510 FOR I = 20 TO 1 STEP - .2 : POKE 54273,I : NEXT
520 POKE 54273,0
525 GOSUB 6000
530 PRINT "{CLR} {DOWN} {DOWN} {RIGHT} {?}BAD ▲LUCK."
535 PRINT "{RIGHT}PLAY"
540 PRINT "{RIGHT}AGAIN?"
545 GET A$ : IF A$ = "Y" THEN POKE 54276,0 :
      POKE 54276,17 : GOSUB 6000 : GOTO 20
550 IF A$ < > "N" THEN GOTO 545
555 GOSUB 6000 : PRINT "{CLR} {DOWN} {DOWN} {DOWN} {RIGHT} {RIGHT} ▄BYE! {DOWN} {DOWN} {DOWN}
        {DOWN} {DOWN} {DOWN}"
560 END
5000 I = - 1 : PRINT "{CLR} {DOWN} {DOWN} {RIGHT} {?} ▲HI ▲I'M"
5005 PRINT "{RIGHT} ▲CALLED"
5010 PRINT "{RIGHT} ▲SIMON"
5020 GOSUB 6000
5025 PRINT "{CLR} {DOWN} {DOWN} {RIGHT} {?} ▲WANT ▲TO"
5030 PRINT "{RIGHT} ▲PLAY ▲MY"
5035 PRINT "{RIGHT} ▲ ▲GAME?"
```

```
5036 GET A$ : IF A$ < > "N" AND A$ < > "Y" THEN 5036
5037 IF A$ = "N" THEN 5500
5038 IF A$ = "Y" THEN GOSUB 6000
5040 PRINT "{CLR}{DOWN}{DOWN}{RIGHT}{COL}REPEAT"
5045 PRINT "{RIGHT}▲▲▲THE"
5050 PRINT "{RIGHT}SEQUENCE"
5055 GOSUB 6000
5060 PRINT "{CLR}{DOWN}{DOWN}{RIGHT}{COL}OF▲LIGHTS"
5065 PRINT "{RIGHT}▲▲▲AND▲"
5070 PRINT "{RIGHT}▲SOUNDS."
5075 GOSUB 6000
5080 PRINT "{CLR}{DOWN}{DOWN}{RIGHT}THERE'S"
5085 PRINT "{RIGHT}MORE▲EACH"
5090 PRINT "{RIGHT}▲▲TIME."
5095 GOSUB 6000
5100 PRINT "{CLR}{DOWN}{DOWN}{RIGHT}{COL}YOU▲MUST"
5105 PRINT "{RIGHT}▲BE▲VERY"
5110 PRINT "{RIGHT}▲▲QUICK!"
5115 GOSUB 6000
5120 PRINT "{CLR}{DOWN}{DOWN}{RIGHT}{COL}▲▲STAND"
5125 PRINT "{RIGHT}▲▲▲BY"
5130 GOSUB 6000
5135 GOSUB 6000
5140 RETURN
5500 GOSUB 6000
5502 PRINT "{CLR}{DOWN}{DOWN}{RIGHT}{COL}BYE▲BYE"
5505 PRINT "{RIGHT}{COL}▲THEN!!{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}"
5510 END
6000 IF I = - 1 THEN FOR Z = 1 TO 2000 : NEXT :
       POKE 54276,0 : POKE 54276,33 : POKE 54273,50
6005 PRINT "{CLR}{DOWN}{DOWN}{RIGHT}▲▲▲▲▲▲▲▲▲▲"
6010 PRINT "{RIGHT}▲▲▲▲▲▲▲▲▲"
6020 PRINT "{RIGHT}▲▲▲▲▲▲▲▲▲"
6025 IF I = - 1 THEN POKE 54273,0
6030 RETURN
9000 DATA 0,0,0,0,255,128,1,255,192,3,255
       ,224,2,0,32,2,0,32,2,54,32,2,99,32
9010 DATA 2,99,32,14,193,184,10,8,40,10,8
       ,40,8,24,8,8,193,136,8,99,8,8,28,8
9020 DATA 4,0,16,4,0,16,7,128,240
       ,5,255,208,2,231,32,99
9030 DATA 2,255,128,10,190,160,42,255,168
       ,170,255,170,174,235,186,174,235,186
9040 DATA 174,235,186,174,235,186,46,170,184
       ,43,170,232,10,235,160,10,235,160
9050 DATA 10,190,160,10,195,160,3,60,192,3
       ,255,192,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
```

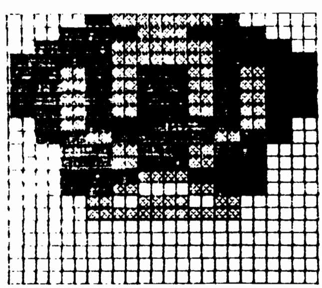
CHEXSUM

```
0=3620
1=3919
2=4728
3=3668
4=1886
5=1157
6=1028
7=1013
8=2145
9=6328
10=769
20=3805
25=1737
30=2133
35=758
40=5486
45=3987
50=2275
51=818
55=351
56=1492
60=1919
62=2997
80=3042
90=2071
91=2394
92=1245
95=1445
98=1321
99=129
100=2094
110=1236
120=2145
130=1878
140=1239
150=2145
160=1878
170=1238
180=2145
190=143
200=2870
210=4524
220=4847
230=4950
240=4528
250=4848
260=4954
270=143
500=1219
510=2028
520=528
525=351
530=1018
535=587
540=691
545=3405
550=1506
555=1448
560=129
5000=1417
5005=749
5010=699
5020=351
5025=959
5030=834
5035=693
5036=2388
5037=1001
5038=1158
5040=982
5045=607
5050=904
5055=351
5060=1052
5065=640
5070=846
5075=351
5080=872
5085=921
5090=699
5095=351
5100=1154
5105=833
5110=777
5115=351
5120=858
5125=532
5130=351
5135=351
5140=143
5500=351
5502=892
5505=897
5510=129
6000=3867
6005=716
6010=625
6020=625
6025=1352
6030=143
9000=3927
9010=3882
9020=2603
9030=4211
9040=4203
9050=4491

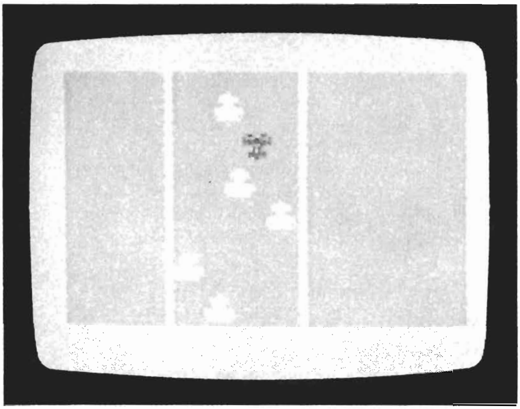
TOTAL= 187570
```

SPRITE

SIMON'S
BODY



Racer



This is a Grand Prix simulation with a different twist.

You have anticipated your win and celebrated before the race, as a result you are drunk and end up driving the race in the wrong direction. Instructions are contained within the program which allows you to select different levels of speed and skill.

Watch out for oil slicks!

PROGRAM STURCTURE	**LINES**
INITIALIZE SPRITE	0 — 9
PRINT LINE OF TRACK ETC	10 — 54
GET KEY PRESSED	100
MOVE CARS	110 — 130
SOMETHING IN ROAD?	140
CRASH ELSE NEXT LINE	200 — 240
CRASH SPIN OUT	500 — 580
THREE CRASHES?	1000
END, ELSE START AGAIN	1000 — 1050
DATA	9000 — 9120

VARIABLES
SP = SPEED
LA = NO. OF LAPS
OS = OBSTACLE (OTHER)
C = OBSTACLE (CAR)
SK = SKILL LEVEL
P = POSITION OF OBSTACLE
L = NO. OF CRASHES
X = YOUR POSITION

RACER

```
0 GOSUB 2000 : POKE 55,255 : POKE 56,48 :
    V = 53248
1 POKE 53281,5 : POKE 53280,7 :
    POKE V + 22, PEEK ( V + 22 ) OR 16 :
    PRINT "{CLR}"
2 POKE 53265,151 : FOR I = 0 TO 111 : READ A :
    POKE 12 * 1024 + I,A : NEXT : POKE V + 24,28
3 POKE 54276,0 : POKE 54296,4 : POKE 54277,132 :
    POKE 54278,240 : POKE 54272,18 : POKE 54273,1
4 POKE 54276,129
5 L = 0 : LA = 0 : X = 150 : POKE V + 34,4 :
    POKE V + 35,7
6 FOR I = 0 TO 62 : READ A : POKE 832 + I,A :
    NEXT : POKE 2040,13 : POKE V + 39,7 :
    POKE V + 37,6
7 POKE V + 38,2 : POKE V + 28,1 : POKE V,X :
    POKE V + 1,100 : POKE V + 21,1
8 FOR I = 12544 TO 12551 : POKE I,0 : NEXT
9 PRINT "{HOME}"; : FOR I = 1 TO 25 : PRINT
    "{10 RIGHT}{WHT}I{12 RIGHT}I" : NEXT : A = PEEK ( V + 31 )
10 LA = LA + .02 : PRINT "{HOME}{24 DOWN}{10 RIGHT}{WHT}I{12 RIGHT}I"
11 IF SP > 1 THEN FOR I = 1 TO SP * 20 : NEXT
12 PRINT "{HOME}{24 DOWN}{10 RIGHT}";
13 IF OS > 0 THEN 21
14 IF C > 0 THEN 51
15 IF INT ( RND ( 1 ) * 10 + 1 ) < 7 + SK THEN 100
16 IF RND ( 1 ) < .95 THEN 50
20 OS = 1 : P = INT ( RND ( 1 ) * 11 + 1 )
21 IF OS = 1 THEN PRINT SPC( P ) ;"JK"; : OS = 2 :
    GOTO 25
22 IF OS = 2 THEN PRINT SPC( P ) ;"LM"; : OS = 0
25 IF C = 0 THEN 100
50 C = 1 : P = INT ( RND ( 1 ) * 10 + 1 )
51 IF C = 1 THEN PRINT SPC( P ) ;"@AB"; : C = 2 :
    GOTO 100
52 IF C = 2 THEN PRINT SPC( P ) ;"CDE"; : C = 3 :
    GOTO 100
54 IF C = 3 THEN PRINT SPC( P ) ;"FGH"; : C = 0
100 PE = PEEK ( 197 )
110 IF PE = 12 THEN X = X - 4
120 IF PE = 36 THEN X = X + 4
130 IF PE = 12 OR PE = 36 THEN POKE V,X :
    POKE V + 1,100
140 IF PEEK ( V + 31 ) = 1 THEN 500
200 IF LA < 12 - ( SK * 2 ) THEN GOTO 10
205 POKE V,0 : POKE 54273,0 : POKE 54272,0
210 POKE V + 24,20 : PRINT "{CLR}{WHT}WELL▲DONE!!!!"
220 PRINT "◣{DOWN}{DOWN}{YEL}YOU'VE▲COMPLETED▲THE▲GAME!"
230 PRINT "{DOWN}▅CRASHES:";L
240 GOTO 1020
```

```
 500 D = - 1 : IF RND ( 1 ) < .6 THEN D = 1
 503 POKE 54272,137 : POKE 54273,43
 505 FOR I = 100 TO 200
 508 POKE 53280,I / 10
 515 X = X + ( D * 4 )
 520 IF X < 118 THEN D = 1
 525 IF X > 185 THEN D = - 1
 530 POKE V,X : POKE V + 1,I
 540 NEXT
 550 X = 150 : POKE V,X : POKE V + 1,100
 560 L = L + 1
 570 IF L = 3 THEN 1000
 575 POKE V + 31,0 : PRINT "{CLR}" : OS = 0 : C = 0 :
       POKE 53280,7
 578 POKE 54272,18 : POKE 54273,1
 580 GOTO 9
1000 POKE V,0 : POKE V + 24,20 :
       PRINT "{CLR} {■}KABOOM!!!!!!"
1002 POKE 54272,43 : POKE 54273,137
1005 POKE 54272,0 : POKE 54273,0 : POKE 54278,0
1006 POKE 54296,15 : POKE 54277,236 : POKE 54278,236
       : POKE 54276,129
1007 POKE 54273,7 : POKE 54272,53
1010 PRINT "{■} {DOWN} {DOWN}YOU▲CRASHED▲THREE▲TIMES!"
1015 FOR I = 1 TO 700 : NEXT : POKE 54278,0 :
       POKE 54272,0 : POKE 54273,0
1020 PRINT "{DOWN} {DOWN} {DOWN} {■}WANT▲ANOTHER▲GO?"
1030 GET A$ : IF A$ = "Y" THEN RUN
1040 IF A$ < > "N" THEN 1030
1050 END
2000 POKE 53281,0
2005 PRINT "{CLR} {■}ROAD▲RACER!!"
2010 PRINT "{DOWN} {■}A▲GAME▲OF▲SKILL.▲DODGE
        ▲THE▲CARS▲AND▲ ▲ ▲ ▲ ▲COMPLETE▲FIFTEEN
        ▲LAPS."
2020 PRINT "{DOWN} {■}YOU▲HAVE▲BEEN▲DRINKING▲TOO▲MUCH
        ▲BEFORE▲ ▲THE▲RACE▲AND▲YOU▲ARE"
2030 PRINT "DRIVING▲THE▲WRONG▲WAY!"
2040 PRINT "{DOWN} {■} {■}AND▲IF▲THATS▲NOT▲ENOUGH▲YOU
        ▲CAN'T▲STOP!"
2050 PRINT "{DOWN}CONTROLS▲ARE:"
2060 PRINT "{DOWN} {■}Z-{■}LEFT"
2070 PRINT "{DOWN} {■}M-{■}RIGHT"
2080 PRINT "{DOWN}DON'T▲CRASH▲INTO▲THE▲SIDE▲OF▲THE
        ▲ROAD!"
2090 PRINT "{DOWN} {DOWN} {■}PRESS▲A▲KEY"
2100 GET A$ : IF A$ = "" THEN 2100
2105 PRINT "{CLR} {■}>>>>>>>>>>>>>>>
       {■}OPTIONS{■}<<<<<<<<<<<<<<<<"
2110 PRINT "{DOWN} {■}SPEED?(1▲FAST▲-▲5▲SLOW)"
2120 GET A$
2130 IF VAL ( A$ ) < 1 OR VAL ( A$ ) > 5 THEN 2120
2140 SP = VAL ( A$ )
2150 PRINT "{DOWN} {■}SKILL▲LEVEL?(1▲HARD▲-▲3▲EASY)"
2160 GET A$
2170 IF VAL ( A$ ) < 1 OR VAL ( A$ ) > 3 THEN 2160
```

```
2180 SK = VAL ( A$ ) ÷ IF SK = 1 THEN SK = 1.5
2190 PRINT "{Q}{Q}{π}GOOD▲LUCK!"
2200 PRINT "{Q}{Q}{■}PRESS▲A▲KEY"
2210 GET A$
2220 IF A$ = "" THEN 2210
2230 RETURN
9000 DATA 0,0,4,7,15,7,4,0,48,48,252,255,255
     ,255,236,236,0,0,64,64,192,64,64,0
9010 DATA 0,0,0,79,127,127,255,79,236,252,204
     ,3,255,255,255,255,0,0,0,196,244,244
9020 DATA 252,196,64,79,15,63,0,0,0,0,48
     ,255,171,255,0,0,0,0
9030 DATA 4,196,192,240,0,0,0,0
9040 DATA 126,126,126,126,60,60,60,60
9050 DATA ,3,15,14,14,58,234,58,0,0,192
     ,192,192,192,252,172
9060 DATA 14,14,58,58,236,243,3,0,171,188
     ,240,192,176,176,176,192
9100 DATA 63,255,240,15,171,192,79,255,196
     ,64,48,4,79,255,196,255,255,252,127,255
9110 DATA 244,127,255,244,79,3,196,0,204
     ,0,0,252,0,0,236,0,0,236,0,4,236,64,7
9120 DATA 255,64,15,255,192,7,255,64,4
     ,252,64,0,48,0,0,48,0,0,0,0
```

CHEXSUM

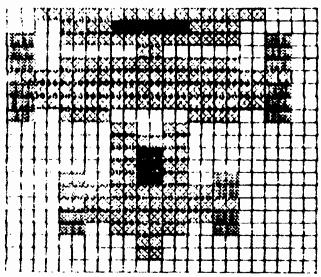
0=2048	51=2515	540=131
1=3375	52=2521	550=1626
2=4062	54=1878	560=561
3=3958	100=796	570=841
4=646	110=1344	575=2545
5=2727	120=1346	578=1212
6=4183	130=2590	580=428
7=3283	140=1393	1000=2631
8=1766	200=1898	1002=1323
9=3984	205=1552	1005=1765
10=3576	210=2161	1006=2825
11=2013	220=2681	1007=1229
12=1373	230=1167	1010=2275
13=814	240=581	1015=2809
14=714	500=1785	1020=1718
15=2224	503=1332	1030=1263
16=1113	505=883	1040=1185
20=1875	508=837	1050=129
21=2606	515=923	2000=530
22=2009	520=1065	2005=1255
25=770	525=1237	2010=5988
50=1774	530=1025	2020=5899

```
2030=1969
2040=3555
2050=1262
2060=909
2070=975
2080=3344
2090=1279
2100=1276
2105=3140
2110=1931
2120=267
2130=2013
2140=737
2150=2431
2160=267
2170=2011
2180=2045
2190=1161
2200=1279
2210=267
2220=906
2230=143
9000=4332
9010=4643
9020=2915
9030=1287
9040=1656
9050=2844
9060=3224
9100=4674
9110=4202
9120=3179

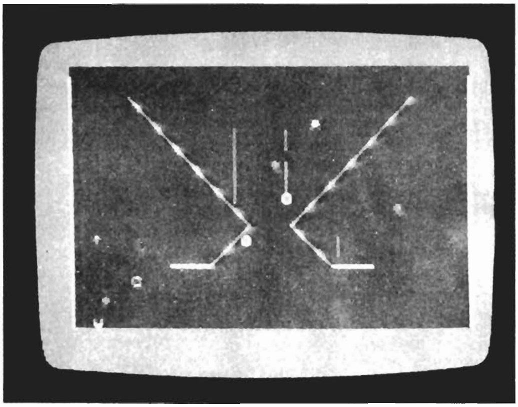
TOTAL= 190694
```

SPRITE

PLAYER'S CAR



Cosmic Bugs



In this game creatures attempt to get from the top of the screen to the bottom where they turn into Crabs which close in on you.

You defend yourself with a blaster that can fire nets or energy blasts. Nets trap the creatures and earn you double points, energy blasts kill them but sometimes only split them in which case you only get half points. If the screen turns yellow you must prepare to move your blaster as it moves up the screen to the top, once successfully at the top the game will resume in a new play area.

The creatures are:

Scorpions — These are red and worth 10 points for a kill.

Spikers — These are green and worth 20 points, or 30 when they leave a spike behind.

Cutters — These are blue and can free other creatures from nets. They are worth 40 points and cannot be netted themselves.

Crabs — These are other creatures who have reached the bottom. Your only method of killing them is to use the super-zapper, (space bar) this uses 25 points. The screen is grey when you have enough points. The first time it is used it kills all life, otherwise only crabs.

Controls :

'Z' blaster left
'C' blaster right
'>' fire nets
'<' fire energy blaster
'space bar' super-zapper

PROGRAM STRUCTURE	LINES
INITIALIZATION	0 — 10
SET VARIABLES: SC, LI, P, NE, N, S, SZ, RE, BO, TO INITIAL VALUES	20
PRINT "FUNNEL" GRAPHICS	25 — 40
GET KEY PRESSED — LEFT OR RIGHT?	50
MOVE LEFT OR RIGHT DEPENDING ON KEY PRESSED	52 — 55
DETERMINE MOVEMENT/ACTIONS OF DIFFERENT ALIEN TYPES	80
TEST FOR CRAB CAPTURED BLASTER	85
ACTION FOR CUTTERS	100 — 112
ACTION FOR SPIKERS	115 — 130
ACTION FOR SCORPIONS	135 — 165
CRABS LANDED!?!	170 — 195
FIRING A NET, FIRING AN ENERGY BLAST	601 — 615
TRAPPED OR HIT AN ALIEN?	615 — 700
MOVE BLASTER UP TO NEW GAME	707 — 725
SPACE BAR HIT TO DESTROY CRABS	805 — 820
BLASTER DESTROYED	1000 — 1020
PLAY AGAIN/HIGH SCORE	1030 — 1055

VARIABLES

SC = SCORE
HI = HIGH SCORE
P(0 — 4) = CONTAINS POSITIONS OF 'BUGS'
T(0 — 4) = CONTAINS TYPE OF 'BUGS'
P = POSITION OF BLASTER
PC = POSITION OF CRAB
L = 'MOVE LEFT'
R = ' MOVE RIGHT'
NE = 'NET'
TI$ = TIME
PO = POSITION OF INDIVIDUAL BUG
CA = ALIEN COUNT
KB = KEYBOARD KEY
K = CHARACTER KEY PRESSED
PE = TEMPORARY
CL = CLASS
NE = NET POSITION
I = VARIABLE
J = VARIABLE
J1 = VARIABLE
V = VIDEO CHIP START ADDRESS

```
  1 POKE 55,255 : POKE 56,47 : V = 53248 :
       POKE V + 33,0 : POKE V + 32,0 : POKE 54296,15
  5 DIM P ( 4 ) ,T ( 4 ) : FOR I = 0 TO 4 :
       T ( I ) = 0 : NEXT : CO = 54272 : KB = 197 :
       LS = 1024 : CA = 0
  7 PC = 11 : L = 12 : R = 20 : HI = 0 :
       FOR I = 12544 TO 12552 : POKE I,0 : NEXT
 10 FOR I = 12288 TO 12415 : READ A : POKE I,A :
       NEXT : POKE V + 24,28
 20 SC = 0 : LI = 3 : P = 2004 : NE = 20 : N = - 1
       : S = - 1 : SZ = 0 : RE = 0 : BO = 0
 25 PRINT "♥ [E] ▲"; : FOR I = 1 TO 24 :
       PRINT "[Q] ■■@"; : NEXT : PRINT "[S] [Q] ■■" :
       FOR I = 1 TO 23 : PRINT "[Q] ■■A";
 30 NEXT : POKE 56295,1 : POKE 2023,1 : PRINT
       "[S] [Q] [Q] [Q] []] []] []] []] []] []]D [Q]D [Q]D [Q]D [Q]D [Q]D
        [Q]D [Q]D [Q]D [Q]D [Q]D [Q]D [Q] ■■";
 35 PRINT "C [Q] ■■ ■■C [Q] ■■ ■■C [Q] ■■ ■■ ■■ ■■ ■■ ■■BBBBC
        ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲DBBBB □ ■■ ■■ ■■ ■■ ■■ ■■D □
       ■■ ■■D □ ■■ ■■D □ ■■C □C □C □C □C □C";
 40 PRINT "□C □C □C □C □C □C"; : TI$ = "000000" :
       POKE 54278,32 : POKE 54277,0 : POKE 54276,0
 45 POKE 54276,17
 50 FOR I = 0 TO 4 : K = PEEK ( KB ) : IF K <> L
       AND K <> R THEN 60
 52 POKE P,32
 55 P = P + ( K = L ) - ( K = R ) :
       IF PEEK ( P ) < > 32 THEN P = P - ( K = L )
       + ( K = R )

 60 POKE P + CO,7 : POKE P,PC
 65 IF T ( I ) > 0 THEN 80
 70 IF RND ( 1 ) < .9 THEN 600
 72 P ( I ) = INT ( LS + RND ( 1 ) * 40 )
 75 T ( I ) = INT ( RND ( 1 ) * 2 + 1 ) :
       IF RND ( 1 ) < CA / 6 THEN T ( I ) = 4
 80 ON T ( I ) GOTO 85,100,115,135,170,185,185,185
 85 CL = 2 : CH = 12 :
       D = INT ( RND ( 1 ) * 3 - 1 ) + 40 :
       PO = P ( I ) + D : PE = PEEK ( PO ) :
       IF PE = PC THEN 1000
 86 IF PE = 5 THEN PE = 32
 87 IF PE = 7 THEN T ( I ) = 6 : POKE P ( I ) ,32 :
       P ( I ) = PO : CA = CA + 1 : SC = SC + 20 :
       GOTO 185
 88 IF PE < > 32 AND RND ( 1 ) > .5 THEN
       D = INT ( RND ( 1 ) * 3 - 1 ) :
       PO = P ( I ) + D : PE = PEEK ( PO ) : GOTO 87
 90 IF PE < > 32 THEN D = 0 : GOTO 500
 95 IF PO > 1983 THEN 130
 97 GOTO 500
100 CL = 5 : CH = 13 :
       D = INT ( RND ( 1 ) * 3 - 1 ) + 40 :
```

```
    PO = P ( I ) + D : PE = PEEK ( PO ) :
    IF PE = PC THEN 1000
101 IF PE = 5 THEN PE = 32
102 IF PE = 7 THEN T ( I ) = 7 : POKE P ( I ) ,32 :
    CA = CA + 1 : P ( I ) = PO : SC = SC + 40 :
    GOTO 185
103 IF PE < > 32 AND RND ( 1 ) > .5 THEN
    D = INT ( RND ( 1 ) * 3 - 1 ) :
    PO = P ( I ) + D : PE = PEEK ( PO ) :
    GOTO 102
105 IF PE < > 32 THEN D = 0 : GOTO 500
110 IF PO > 1983 THEN 130
111 IF RND ( 1 ) > .9 AND D > 0 THEN T ( I ) = 3
112 GOTO 500
115 POKE P ( I ) ,5 : CL = 5 : CH = 13 : D = 40 :
    PO = P ( I ) + D : PE = PEEK ( PO ) :
    IF PE = PC THEN 1000
116 IF PE = 7 THEN T ( I ) = 8 : POKE P ( I ) ,32 :
    P ( I ) = PO : CA = CA + 1 : SC = SC + 60 :
    GOTO 185
117 IF PE = 5 THEN PE = 32
120 IF PE < > 32 THEN T ( I ) = 0 : GOTO 600
125 IF PO < 1984 THEN P ( I ) = P ( I ) + D :
    GOTO 505
130 T ( I ) = 5 : POKE P ( I ) ,32 :
    P ( I ) = P ( I ) + D : POKE P ( I ) + CO,10
    : POKE P ( I ) ,15 : GOTO 600
135 CL = 14 : CH = 14 :
    D = INT ( RND ( 1 ) * 3 - 1 ) + 40 :
    PO = P ( I ) + D : PE = PEEK ( PO ) :
    IF PE = PC THEN 1000
136 IF PE = 5 THEN PE = 32
140 IF PE < > 8 THEN 153
145 FOR J = 0 TO 4 : IF P ( J ) = PO AND
    T ( J ) > 5 THEN T ( J ) = T ( J ) - 5 :
    D = 0 : J = 4 : CA = CA - 1 : NE = NE + 1
150 NEXT J : GOTO 500
152 IF PE < > 32 AND RND ( 1 ) > .5 THEN
    D = INT ( RND ( 1 ) * 3 - 1 ) :
    PO = P ( I ) + D : PE = PEEK ( PO ) : GOTO 136
153 IF PE < > 32 AND RND ( 1 ) > .5 THEN
    D = INT ( RND ( 1 ) * 3 - 1 ) :
    PO = P ( I ) + D : PE = PEEK ( PO )
155 IF PE < > 32 THEN D = 0 : GOTO 500
160 IF PO > 1983 THEN POKE P ( I ) ,32 :
    T ( I ) = 0 : GOTO 600
165 GOTO 500
170 CL = 10 : CH = 15 :
    D = INT ( RND ( 1 ) * 3 - 1 ) :
    IF RND ( 1 ) > .1 THEN
    D = ( P ( I ) > P ) - ( P ( I ) < P )
175 PO = P ( I ) + D : PE = PEEK ( PO ) :
    IF PE = PC THEN 1000
180 IF PE < > 32 THEN D = 0
182 GOTO 500
185 CL = INT ( RND ( 1 ) * 15 + 1 ) : CH = 8 : D = 0
```

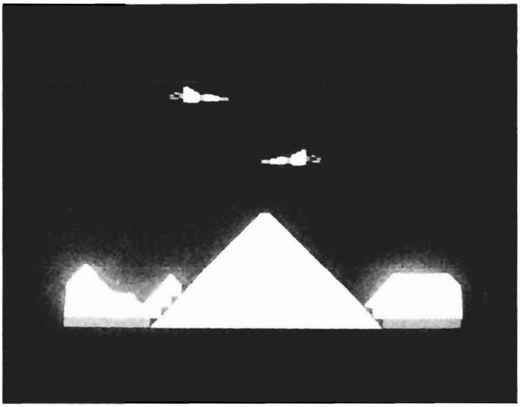
```
190 IF CA = 5 THEN POKE P ( J ) ,32 : T ( J ) = 0 :
     CA = CA - 1 : NE = NE + 1 : GOTO 600
195 GOTO 505
500 POKE P ( I ) ,32 : P ( I ) = P ( I ) + D
505 POKE P ( I ) + CO,CL : POKE P ( I ) ,CH
600 IF K < > 44 OR N > - 1 OR NE = 0 THEN 605
601 N = P : NE = NE - 1 : POKE 54290,0 :
     POKE 54291,71 : POKE 54287,34 : POKE 54286,75
602 POKE 54290,17
605 IF K < > 47 OR S > - 1 THEN 610
606 S = P : POKE 54290,0 : POKE 54291,71 :
     POKE 54287,72 : POKE 54286,169 : POKE 54290,33
610 IF N = - 1 THEN 650
615 POKE N,32 : N = N - 40 : IF N < LS THEN N = - 1
     : GOTO 650
617 IF RND ( 1 ) > .95 THEN N = N + 40 :
     POKE N + CO,4 : POKE N,7 : N = - 1 : GOTO 650
620 IF PEEK ( N ) = 32 THEN POKE CO + N,4 :
     POKE N,6 : GOTO 650
640 N = N + 40 : POKE N + CO,4 : POKE N,7 : N = - 1
650 IF S = - 1 THEN 700
655 POKE S,32 : S = S - 40 : IF S < LS THEN S = - 1
     : GOTO 700
660 PE = PEEK ( S ) : C = PEEK ( S + CO ) :
     POKE S + CO,8 : POKE S,9 : IF PE = 32 THEN 700
661 POKE 54290,0 : POKE 54287,8 : POKE 54286,37 :
     POKE 54291,36 : POKE 54290,129
665 POKE S + CO,7 : POKE S,10 : FOR J = 0 TO 4 :
     IF T ( J ) = 0 OR P ( J ) < > S THEN 680
670 IF T ( J ) > 5 THEN T ( J ) = T ( J ) - 5 :
     SC = SC + 5 : J = 100 : CA = CA - 1 :
     NE = NE + 1 : GOTO 680
675 SC = SC + T ( J ) * 10 : IF RND ( 1 ) < .7
     THEN T ( J ) = 0 : J = 100 : GOTO 680
677 FOR J1 = 0 TO 4 : IF T ( J1 ) = 0 THEN
     T ( J1 ) = T ( J ) : P ( J1 ) = P ( J ) :
     J1 = 100 : J = 100
678 NEXT J1 : IF J1 < 99 THEN T ( J ) = 0 : J = 100
680 NEXT J : IF J > 5 THEN POKE S,32 : GOTO 690
682 IF PE = 7 THEN NE = NE + 1 : PE = 32
683 IF PE = 5 THEN PE = 32
685 POKE S + CO,C : POKE S,PE
690 S = - 1
700 POKE 54273, RND ( 1 ) * 50 : POKE 54272,
     RND ( 1 ) * 256 : IF K = 60 THEN 800
701 RE = 0 : IF SC < 25 THEN POKE V + 32,0
702 IF TI$ > "000250" THEN POKE V + 32,7 : GOTO 704
703 IF SC > = 25 THEN POKE V + 32,11
704 IF BO = 0 AND SC > 1000 THEN SZ = 0 : BO = 1
705 NEXT I : IF TI$ < "000300" THEN 50
707 IF S > - 1 THEN POKE S,32 : S = - 1
708 IF N > - 1 THEN POKE N,32 : N = - 1
710 FOR I = 0 TO 4 : IF T ( I ) > 0 THEN
     POKE P ( I ) ,32 : T ( I ) = 0
715 NEXT : NE = 20 : RE = 0 : SZ = 0 : CA = 0
717 POKE V + 33,7 : FOR I = 1 TO 1000 : NEXT :
```

```
         POKE V + 33,0
  720 K = PEEK ( KB ) ÷ POKE P,32 ÷
         P = P - 40 + ( K = L ) - ( K = R )
  725 PE = PEEK ( P ) ÷ POKE P + CO,7 ÷ POKE P,PC ÷
         IF P < LS THEN P = 2004 ÷ SC = SC + 100 ÷
         GOTO 25
  727 FOR I = 1 TO 50 ÷ NEXT
  730 IF PE = 32 THEN 720
  735 LI = 0 ÷ I = 0 ÷ P ( I ) = P ÷ GOTO 1000
  800 IF SC < 25 OR RE THEN 705
  805 POKE V + 33,13 ÷ SC = SC - 25 ÷ FOR J = 0 TO 4
         ÷ IF T ( J ) = 0 THEN 820
  806 POKE 54290,0 ÷ POKE 54287,72 ÷ POKE 54286,169 ÷
         POKE 54291,17 ÷ POKE 54290,129
  810 IF SZ = 1 AND TJ < > 5 THEN 820
  811 IF T ( J ) > 5 THEN CA = CA - 1 ÷ NE = NE + 1
  815 POKE P ( J ) ,32 ÷ T ( J ) = 0
  820 NEXT J ÷ SZ = 1 ÷ POKE V + 33,0 ÷ RE = 1 ÷
         GOTO 705
 1000 POKE P ( I ) ,32 ÷ POKE V + 33,11 ÷
         FOR J = 1 TO 1000 ÷ NEXT
 1002 POKE 54290,0 ÷ POKE 54287,4 ÷ POKE 54286,73 ÷
         POKE 54291,240 ÷ POKE 54290,129
 1005 POKE V + 33,12 ÷ FOR J = 1 TO 1000 ÷ NEXT
 1010 POKE V + 33,15 ÷ FOR J = 1 TO 1000 ÷ NEXT
 1015 POKE V + 33,0 ÷ FOR J = 1 TO 1000 ÷ NEXT
 1020 T ( I ) = 0 ÷ SZ = 0 ÷ LI = LI - 1 ÷
         IF LI > 0 THEN 700
 1025 POKE V + 24,21 ÷ PRINT "♥πGAME OVER" ÷
         PRINT "QQQQYOU SCORED";SC
 1030 IF SC > HI THEN HI = SC
 1035 PRINT "≡QQQHI IS";HI
 1040 PRINT "QQQ▓ANOTHER GAME?"
 1042 FOR I = 0 TO 4 ÷ T ( I ) = 0 ÷ P ( I ) = 0 ÷
         NEXT
 1045 GET A$ ÷ IF A$ = "Y" THEN RESTORE ÷ GOTO 10
 1050 IF A$ < > "N" THEN 1045
 1052 POKE 54272,0 ÷ POKE 54273,0 ÷ POKE 54287,0 ÷
         POKE 54286,0
 1055 END
10000 DATA 128,128,128,128,128,128,128,128,1,1
         ,1,1,1,1,1,1,0,0,0,0,0,0,0,255
10005 DATA 1,2,4,8,16,32,64,128,128,64,32,16,8
         ,4,2,1,8,8,8,8,8,8,8,8
10010 DATA 8,28,62,28,8,0,8,8,62,107,73,119
         ,73,107,62,0,62,65,65,73,65,65,62,0
10015 DATA 8,28,42,42,8,0,8,0,65,34,54,60,252
         ,60,34,65,0,65,65,65,107,107,62,28
10020 DATA 56,214,124,214,56,16,40,68,0,146
         ,124,56,254,56,68,130
10025 DATA 0,16,40,68,130,68,40,16,0,68
         ,56,84,254,124,130,68
```

CHEXSUM

0=0	135=6098	685=1196	10005=3289
1=3793	136=1163	690=485	10010=4222
5=4738	140=1062	700=3346	10015=4315
7=3757	145=6107	701=1879	10020=3113
10=2899	150=802	702=2432	10025=2887
20=4733	152=5927	703=1645	
25=4275	153=5272	704=2346	
30=4119	155=1866	705=1545	TOTAL= 345191
35=8548	160=2715	707=1772	
40=4524	165=529	708=1738	
45=591	170=5387	710=2764	
50=3247	175=2930	715=2047	
52=390	180=1263	717=2467	
55=4610	182=529	720=3232	
60=1180	185=2323	725=4964	
65=918	190=4218	727=908	
70=1110	195=529	730=930	
72=1667	500=1631	735=2032	
75=3428	505=1666	800=1291	
80=2468	600=2283	805=3418	
85=6024	601=3842	806=3372	
86=1163	602=597	810=1682	
87=5129	605=1735	811=2378	
88=5980	606=3769	815=1130	
90=1666	610=964	820=2512	
95=1072	615=3084	1000=2487	
97=529	617=4044	1002=3339	
100=6024	620=2712	1005=1794	
101=1163	640=2362	1010=1797	
102=5152	650=964	1015=1737	
103=5925	655=3117	1020=2829	
105=1866	660=4070	1025=3777	
110=1072	661=3290	1030=1294	
111=1945	665=3964	1035=884	
112=529	670=5598	1040=1350	
115=5192	675=3915	1042=2024	
116=5127	677=4478	1045=1809	
117=1163	678=2054	1050=1182	
120=2069	680=1936	1052=2392	
125=2562	682=1954	1055=129	
130=4680	683=1163	10000=3940	

Star Duel



Ram challenges you to a dog fight. The game begins when you accept the challenge.

You must avoid the ground, his craft and his missiles while trying to manoeuvre your craft into a position which will enable you to fire your missiles at him.

Your craft is controlled by using the 'W' and 'X' keys to move up and down, and the 'A' and 'D' keys to shift left and right. When you have Ram in your sights use the 'S' key to fire your missile.

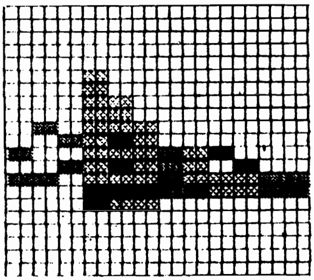
Remember, if either craft is hit it will crash to the ground.

Yours is the blue craft. See if you can outsmart him!

PROGRAM STUCTURE	LINES
INITIALIZATION	0 — 25
CHANGE DIRECTION OF OPPONENT (OR ON COLLISION)	26
CALCULATE NEW POSITION AND MOVE OPPONENT	27 — 48
IF HIT (YOU): DROP, MUSIC AND MESSAGES	49
GET KEY PRESSED, IF HIT OPPONENT: DROP, MUSIC AND MESSAGES	50
CHANGE DIRECTION OR HEIGHT ACCORDING TO KEY	55 — 70
IF 'FIRE KEY' PRESSED	72
POSITION BULLET	73
NOISE	74
MOVE IN DIRECTION D	75
WRAP AROUND SCREEN	80 — 101
IF HIT (YOU): DROP, MUSIC AND MESSAGES	102 — 104
IF BULLET TRAVELLING	110
MOVE	120 — 131
IF IT HITS: DROP, MUSIC AND MESSAGES	132
IF OPPONENT BULLET TRAVELLING	150
MOVE	170 — 181
IF IT HITS: DROP, MUSIC AND MESSAGES	182
GO TO CHANGE DIRECTION OF OPPONENT	

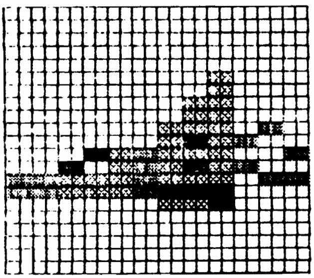
DROP ROUTINE (YOU)	1000 — 1020
DROP ROUTINE (OPPONENT)	2000 — 2020
MUSIC AND MESSAGE ROUTINE (OPPONENT)	5000 — 5040
MUSIC AND MESSAGE ROUTINE (YOU)	6000 — 6040
SPRITE DATA	9000 — 9110

VARIABLES
X, Y = YOUR POSITION
X1, Y1 = OPPONENT'S POSITION
D = YOUR DIRECTION
D1 = OPPONENT'S DIRECTION
SX = BULLET POSITION
SD = BULLET DIRECTION
SZ = OPPONENT'S BULLET POSITION
DZ = OPPONENT'S BULLET DIRECTION



SPRITE

PLAYER/ALIEN
SHIP
MOVING
RIGHT



SPRITE

PLAYER/ALIEN
SHIP
MOVING
LEFT

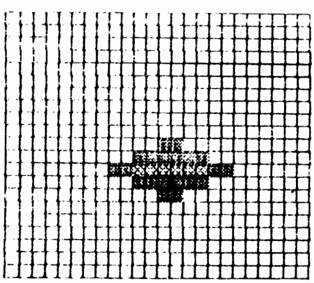
STAR DUEL

```
 0 POKE 54296,10 : POKE 54277,0 : POKE 54278,240 :
     POKE 54276,0 : POKE 54276,129 : SX = 0
 1 POKE 54272,0 : POKE 54273,0 : PRINT "{CLR}" :
     POKE 53280,4 : POKE 53281,4 : CO = 54272
 2 V = 53248 : POKE 2040,192 : POKE V + 39,14 :
     POKE 2041,193 : POKE 2042,194 : POKE 2043,194
 3 POKE 2044,195 : POKE V + 40,5 : POKE V + 43,0 :
     POKE V + 28,31 : POKE V + 37,2 : POKE V + 38,7
 4 FOR I = 0 TO 254 : READ Q : POKE 12288 + I,Q :
     NEXT : POKE V + 21,31 : POKE V + 29,19 :
     POKE V,100
 5 POKE V + 2,200 : POKE V + 4,50 :
     POKE V + 6,230 : POKE V + 8,165
 6 GOSUB 4000
 7 PE = PEEK ( V + 30 ) : PE = PEEK ( V + 31 ) :
     X = 100 : Y = 60 : D = 4 : X1 = 200 :
     Y1 = 120 : D1 = - 4
10 PRINT "{HOME}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}
     {DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{RIGHT}{RVS}◢◣{DOWN}▮▮▮▮▮▮◢▲▲◣{RIGHT}{RIGHT}{RIGHT}
     {RIGHT}{RIGHT}{RIGHT}◢◣{RIGHT}{RIGHT}{RIGHT}{RIGHT}◢▲◣{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}◢
     ◣{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}◢▲▲▲▲▲▲◣";
12 PRINT "{RVS}▲▲▲▲◣{RIGHT}{RIGHT}{RIGHT}{RIGHT}◢▲▲◣{RIGHT}{RIGHT}
     ◢▲▲▲◣{RIGHT}{RIGHT}{RIGHT}{RIGHT}◢▲▲◣{RIGHT}{RIGHT}{RIGHT}◢
     ▲▲▲▲▲▲▲▲";
14 PRINT "{RVS}▲▲▲▲▲▲▲◣◢▲▲▲▲◣
     ◢▲▲▲▲▲◣{RIGHT}{RIGHT}◢▲▲▲▲◣{RIGHT}◢
     ▲▲▲▲▲▲▲▲▲";
15 PRINT "{RVS}▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲
     ▲▲▲◣◢▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲
     {HOME}" : TA = INT ( RND ( 1 ) * 10 )
16 FOR I = 56256 TO 56295 : POKE I - CO,160 :
     POKE I,9 : NEXT
18 FOR I = 56256 + TA TO 55837 + TA STEP - 39 :
     POKE I - CO,233 : POKE I,5 : NEXT
19 FOR I = 56278 + TA TO 55837 + TA STEP - 41 :
     POKE I - CO,223 : POKE I,5 : NEXT
20 FOR I = 0 TO 10 : FOR J =
     56257 - I * 40 + I + TA TO 56277 - I * 40 - I
     + TA : POKE J - CO,160 : POKE J,5 : NEXT
25 NEXT
26 IF RND ( 1 ) < .05 OR PEEK ( V + 30 ) = 2
     THEN D1 = - D1 : POKE 54273,10
27 X1 = X1 + D1 : IF RND ( 1 ) < .1 THEN
     Y1 = Y1 + ( ( Y1 > Y ) - ( Y1 < Y ) ) * 4 :
     GOTO 30
28 Y1 = Y1 + ( INT ( RND ( 1 ) * 3 ) - 1 ) * 4
30 IF X1 < 10 AND D1 = - 4 THEN X1 = 320
35 IF X1 > 320 AND D1 = 4 THEN X1 = 0
36 IF Y1 < 60 THEN Y1 = 60
40 IF Y1 > 200 THEN Y1 = 200
41 IF RND ( 1 ) > .2 OR Y1 < Y - 10 AND
     Y1 > Y + 10 AND RND ( 1 ) < .30 OR
     S2 < > 0 OR X1 > 280 THEN 45
43 S2 = X1 + ( D1 * 5 ) : D2 = D1 * 3 :
```

```
      IF D1 = 4 THEN S2 = S2 + 30 : POKE V + 7,Y1
  44 FOR I = 50 TO 0 STEP - 2 : POKE 54273,I :
      NEXT : POKE 54273,0
  45 IF D1 = - 4 THEN POKE 2041,193
  46 IF D1 = 4 THEN POKE 2041,192
  47 POKE V + 3,Y1 : IF X1 > 255 THEN
      POKE V + 2,X1 - 256 : POKE V + 16,
      PEEK ( V + 16 ) OR 2 : GOTO 49
  48 POKE V + 2,X1 : POKE V + 16, PEEK ( V + 16 )
      AND 253
  49 IF PEEK ( V + 30 ) = 3 THEN 1000
  50 PE = PEEK ( 197 ) : IF PEEK ( V + 30 ) = 6 THEN 2000
  55 IF PE = 9 AND Y > 60 THEN Y = Y - 4 :
      POKE 54273,200
  60 IF PE = 23 THEN Y = Y + 4 : POKE 54273,200
  65 IF PE = 10 THEN D = - 4 : POKE 54273,10
  70 IF PE = 18 THEN D = 4 : POKE 54273,10
  72 IF PE < > 13 OR SX < > 0 OR X > 200 THEN 75
  73 SX = X + ( D * 5 ) : POKE V + 5,Y : SD = D * 3
      : IF D = 4 THEN SX = SX + 30
  74 FOR I = 50 TO 0 STEP - 2 : POKE 54273,I :
      NEXT : POKE 54273,0
  75 X = X + D : IF D = 4 THEN POKE 2040,192
  80 IF D = - 4 THEN POKE 2040,193
  90 IF X < 10 AND D = - 4 THEN X = 320
  95 IF X > 320 AND D = 4 THEN X = 0
 100 POKE 54273,0 : POKE V + 1,Y : IF X > 255 THEN
      POKE V,X - 256 : POKE V + 16, PEEK ( V + 16 )
      OR 1 : GOTO 102
 101 POKE V,X : POKE V + 16, PEEK ( V + 16 ) AND 254
 102 IF PEEK ( V + 300 ) = 3 THEN 1000
 103 IF PEEK ( V + 30 ) = 9 THEN 1000
 104 IF PEEK ( V + 31 ) = 1 THEN 1000
 110 IF SX = 0 THEN 150
 120 IF SX < 10 OR SX > 310 THEN SX = 0 : SD = 0 :
      POKE V + 5,0 : POKE V + 4,50
 130 SX = SX + SD : IF SX > 255 THEN
      POKE V + 4,SX - 256 : POKE V + 16,
      PEEK ( V + 16 ) OR 4 : GOTO 132
 131 POKE V + 4,SX : POKE V + 16, PEEK ( V + 16 )
      AND 251
 132 IF PEEK ( V + 30 ) = 6 THEN 2000
 150 IF S2 = 0 THEN 26
 170 IF S2 < 10 OR S2 > 310 THEN S2 = 0 : D2 = 0 :
      POKE V + 7,0 : POKE V + 6,230
 180 S2 = S2 + D2 : IF S2 > 255 THEN
      POKE V + 6,S2 - 256 : POKE V + 16,
      PEEK ( V + 16 ) OR 8 : GOTO 182
 181 POKE V + 6,S2 : POKE V + 16, PEEK ( V + 16 )
      AND 247
 182 IF PEEK ( V + 30 ) = 9 THEN 1000
 190 GOTO 26
1000 POKE V + 6,0 : POKE V + 7,0 : FOR I = Y TO 200
      : POKE V + 1,I : POKE 54273,I
1005 FOR J = 1 TO 5 : NEXT : NEXT : POKE 54273,0
1010 FOR I = 1 TO 7 : POKE V + I,0 : NEXT
```

```
1020 GOTO 5000
2000 POKE V + 4,0 : POKE V + 5,0 :
       FOR I = Y1 TO 200 : POKE V + 3,I :
       POKE 54273,I
2005 FOR J = 1 TO 5 : NEXT : NEXT : POKE 54273,0
2010 FOR I = 1 TO 7 : POKE V + I,0 : NEXT
2020 GOTO 6000
4000 PRINT "{S}{E}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}
       {■}STAR▲DUEL{E}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}";
4001 FOR I = 1 TO 11
4002 PRINT "{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}
       {]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}
       {]}{]}{]}{]}{]}";
4003 NEXT
4004 POKE V + 8,165 : POKE V + 9,100
4010 PRINT "{]}{■}▲ ▲I▲AM▲RAM▲OF▲PLANET
       ▲BYTE. ▲I▲HAVE▲CHOSEN▲ ▲YOU▲TO▲DEFEND
       ▲YOUR▲PLANET."
4020 PRINT "{E}I▲CHALLENGE▲YOU▲TO▲A▲DUEL!"
4025 PRINT "DO▲YOU▲ACCEPT?"
4026 GET A$ : IF A$ = "N" THEN PRINT "{R}COWARD" :
       END
4028 IF A$ < > "Y" THEN 4026
4030 PRINT "▲ ▲THE▲DUEL▲WILL▲BE▲ABOVE▲THE
       ▲PLANET▲MARS"
4040 PRINT "{□}▲ ▲{E}I▲MUST▲WARN▲YOU▲THAT▲ONLY
       ▲ONE▲OF▲YOUR▲ ▲ ▲SHOTS▲CAN▲DESTROY▲ME."
4050 PRINT "{■}▲ ▲YOU▲HOWEVER▲CAN▲CRASH!"
4060 PRINT "{E}{Q}▲ ▲ ▲ ▲PRESS▲'SPACE'▲WHEN▲YOU
       ▲ARE▲READY."
4065 GET A$ : IF A$ < > "▲" THEN 4065
4066 PRINT "{♥}" : POKE V + 9,0
4067 POKE 53281,0 : POKE 53280,0
4070 RETURN
5000 POKE 53280,4 : POKE 53281,4 : PRINT "{♥}" :
       POKE 54273,0
5001 PRINT "{S}"; : FOR I = 1 TO 12
5002 PRINT "{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}
       {]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}{]}
       {]}{]}{]}{]}{]}{]}";
5003 NEXT
5010 PRINT "{■}GOODBYE,MORTAL.YOU▲FOUGHT▲WELL."
5020 PRINT "{E}{Q}{Q}{Q}DOES▲ANOTHER▲HUMAN▲DARE
       ▲TO▲CHALLENGE▲ME?"
5025 GET A$ : IF A$ = "Y" THEN GOSUB 4066 : GOTO 7
5030 IF A$ < > "N" THEN 5025
5035 PRINT "{■}VERY▲WELL,THE▲EARTH▲WILL▲SOON▲BE
       ▲MINE!"
5040 END
6000 POKE V + 23,16 : POKE 53280,4 : POKE 53281,4 :
       POKE V + 8,150 : POKE V + 9,100 : POKE 54273,0
6005 POKE 54296,0 : POKE 54296,15 : POKE 54276,0 :
       POKE 54276,17
6010 PRINT "{E}{S}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}I▲AM
       ▲DYING▲EARTH▲MAN!"
6020 FOR I = 1 TO 200 : POKE 54273,I : NEXT :
```

```
       POKE V + 9,0 : FOR I = 100 TO 1 STEP - .5
6025 POKE 54273,I : NEXT : FOR I = 10 TO 50 STEP .5
6026 POKE 54273,I : NEXT : POKE V + 23,0 :
       POKE V + 29,0 : POKE V + 8,150 :
       POKE V + 9,100
6028 FOR I = 1 TO 50 STEP .5 : POKE 54273,I : NEXT :
       POKE V + 8,0 : POKE V + 9,0 : FOR I = 20 TO 1
       STEP - .2
6029 POKE 54273,I : NEXT : POKE 54273,0
6030 PRINT "▄ ▣GOODBYEEE!!!!"
6035 GET A$ : FOR I = 1 TO 5000 : NEXT : PRINT "▣"
6040 END
9000 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,3
       ,0,0,3,0,0,3,192,0,3,192,0,19,240,0
9010 DATA 7,176,0,67,247,128,7,183,32,83,247
       ,245,2,166,245,2,240,0
9020 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,99
9030 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
       ,0,192,0,0,192,0,3,192,0,3,192,0,15,196
9040 DATA 0,14,208,2,223,193,8,222,208,95
       ,223,197,95,218,128,0,15,128,0,0,0,0,0
9050 DATA 0,0,0,0,0,0,0,0,0,0,99
9060 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
       ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,4,0
9070 DATA 0,21,0,0,127,64,0,21,0,0,4,0,0,0
       ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,99
9090 DATA 60,0,60,12,170,48,15,170,240,14
       ,170,176,2,130,128,2,0,128,2,60,128
9100 DATA 10,60,160,10,0,160,10,0,160,8,0
       ,32,8,170,32,8,150,32,8,150,32,88,170
9110 DATA 37,88,0,37,90,0,165,21,170,84,21
       ,130,84,5,170,80,5,170,80
```



SPRITE

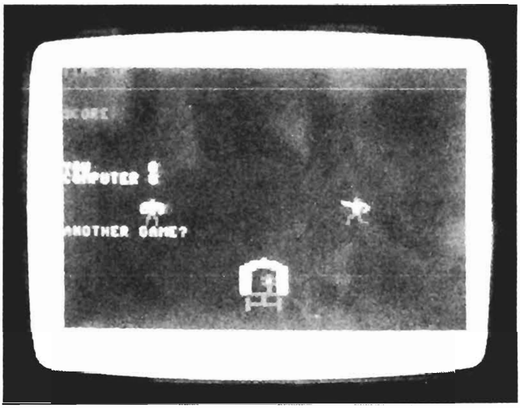
PLAYER/ALIEN
MISSILE

CHEXSUM

0=3852
1=3668
2=4070
3=4327
4=4405
5=2938
6=351
7=5336
10=6009
12=3253
14=3399
15=3902
16=2693
18=3903
19=3897
20=6172
25=131
26=3417
27=4676
28=2084
30=1957
35=1742
36=1195
40=1321
41=5265
43=4257
44=2626
45=1460
46=1280
47=4843
48=2376
49=1456
50=2310
55=2610
60=2067
65=1917
70=1756
72=2475
73=4121
74=2626
75=1926
80=1401
90=1811
95=1592
100=5151
101=2048
102=1498
103=1456
104=1456
110=888
120=3812
130=5130
131=2405
132=1460
150=792
170=3731
180=4966
181=2372
182=1456
190=474
1000=3482
1005=1676
1010=1528
1020=583
2000=3561
2005=1676
2010=1528
2020=583
4000=8393
4001=711
4002=9525
4003=131
4004=1438
4010=6673
4020=2244
4025=1291
4026=2129
4028=1205
4030=3587
4040=6133
4050=2222
4060=3186
4065=1492
4066=1005
4067=1171
4070=143
5000=2305
5001=1122
5002=9525
5003=131
5010=2931
5020=3771
5025=2007
5030=1188
5035=3502
5040=129
6000=4161
6005=2507
6010=2425
6020=3641
6025=1914
6026=3755
6028=4508
6029=1365
6030=1302
6035=1864
6040=129
9000=4013
9010=3247
9020=1896
9030=4490
9040=4429
9050=1307
9060=3865
9070=4137
9090=4079
9100=4299
9110=3292

TOTAL= 328500

Cowboy Shootout



Shoot the computer-controlled cowboy as many times as you can without him shooting you. As you shoot it out you have to avoid the wagons passing between you. You can't shoot through them.

Your controls are:

- 'W' for up
- 'X' for down
- 'S' to fire

You're on the left, the computer is on the right.

Straight shooting!

PROGRAM STRUCTURE	LINES
INITIALIZATION	1—45
GET KEYBOARD CHARACTER	50
PUT PLAYER AT RESULTANT POSITION	55, 57
PUT COMPUTER	60—80
PLAYER SHOOTS	85, 90, 130, 135
COMPUTER SHOOTS	100, 125, 155, 160
MOVE CARRIAGE	165
SHRINK CARRIAGE	170
TEST IF PLAYER SHOT	175
TEST IF COMPUTER SHOT	176
TIME DISPLAY	182
DATA	10000—10115

VARIABLES
K = KEYBOARD CHARACTER

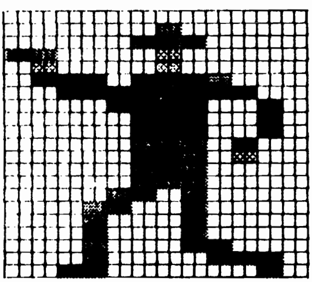
A = ALTERNATELY HOLDS THE POSITIONS OF COMPUTER AND PLAYER
PS = PLAYER SHOOT FLAG
CS = COMPUTER SHOOT FLAG

COWBOY SHOOT-OUT

```
  1 POKE 54296,0 : POKE 55,255 : POKE 56,47
      : V = 53248 : POKE 54296,15
  5 FOR I = 0 TO 510 : READ A : POKE 12 * 1024 + I,A
      : NEXT : PRINT "♥"
 10 POKE 2040,192 : POKE 2041,192 : POKE V,86
      : POKE V + 1,50 : POKE V + 2,244
      : POKE V + 3,200
 15 POKE V + 39,14 : POKE V + 40,7 : POKE V + 33,8
      : POKE V + 32,5 : POKE V + 28,19
      : POKE V + 37,9
 20 POKE V + 38,10 : POKE V + 21,19 : POKE V + 41,1
      : POKE V + 42,1 : POKE 2042,198 : POKE 2043,198
 25 POKE 2044,199 : POKE V + 43,7 : POKE V + 8,160
      : POKE V + 9,248 : POKE V + 23,16
      : POKE V + 29,16
 45 KB = 197 : PS = 0 : CS = 0 : PC = 0 : CC = 0
      : A = PEEK ( V + 30 ) : CO = 0 : YO = 0
      : TI$ = "000000"
 50 K = PEEK ( KB ) : A = PEEK ( V + 1 )
      : A = A + ( K = 23 ) * ( A < 225 )
        * 2 - ( K = 9 ) * ( A > 50 ) * 2
 55 POKE V + 1,A : PC = PC + 1
      : IF PC = 3 THEN POKE 2040,192 : PC = 0
 57 CC = CC + 1 : IF CC = 3 THEN POKE 2041,192
      : CC = 0
 60 A = PEEK ( V + 3 )
      : IF RND ( 1 ) > .3
        THEN A = A + INT ( RND ( 1 ) * 3 - 1 ) * 2
      : GOTO 70
 65 A = A + ( A > PEEK ( V + 1 ) )
      * 2 - ( A < PEEK ( V + 1 ) ) * 2
 70 IF A > 255 THEN A = 255
 75 IF A < 50 THEN A = 50
 80 POKE V + 3,A : IF K < > 13 OR PS THEN 100
 85 POKE 2040,194 : PC = 0 : PS = 1 : POKE V + 4,98
      : POKE V + 5, PEEK ( V + 1 )
      : POKE V + 21, PEEK ( V + 21 ) OR 4
 90 POKE 54273,17 : POKE 54272,37 : POKE 54277,24
      : POKE 54276,0 : POKE 54276,129
100 IF RND ( 1 ) < .9 OR CS THEN 125
105 POKE 2041,193 : CC = 0 : CS = 1 : POKE V + 6,232
      : POKE V + 7, PEEK ( V + 3 )
      : POKE V + 21, PEEK ( V + 21 ) OR 8
110 POKE 54287,17 : POKE 54286,37 : POKE 54291,24
      : POKE 54290,0 : POKE 54290,129
125 IF PS = 0 THEN 150
130 A = PEEK ( V + 4 ) + 8
      : IF A > 255
        THEN POKE V + 21, PEEK ( V + 21 ) - 4
      : PS = 0 : GOTO 150
135 POKE V + 4,A
150 IF CS = 0 THEN 165
155 A = PEEK ( V + 6 ) - 8
      : IF A < 80
```

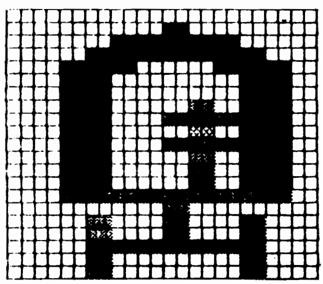
```
         THEN POKE V + 21, PEEK ( V + 21 ) - 8
       : CS = 0 : GOTO 175
 160 POKE V + 6,A
 165 POKE V + 9, PEEK ( V + 9 ) - 4
       : IF PEEK ( V + 9 ) = 8 THEN POKE V + 23,16
       : POKE V + 29,16 : POKE V + 9,244
 170 IF PEEK ( V + 9 ) = 120 THEN POKE V + 23,0
       : POKE V + 29,0
 175 A = PEEK ( V + 30 ) : IF A = 9 THEN 1000
 176 IF A = 6 THEN 2000
 177 IF A = 20 THEN PS = 0 : POKE V + 4,0
 180 IF A = 24 THEN CS = 0 : POKE V + 6,0
 182 PRINT "{HOME}{RIGHT}";TI$
 185 IF TI$ < "000100" THEN 50
 190 POKE V + 33,0 : POKE V + 27,255
       : PRINT "{CLR}{RED}TIME UP" : PRINT "{DOWN}{DOWN}{DOWN}SCORE:"
 195 PRINT "{RIGHT}{DOWN}{DOWN}{DOWN}{DOWN}YOU      ";YO
       : PRINT "COMPUTER";CO
 200 PRINT "{DOWN}{DOWN}{DOWN}{DOWN}{GREEN}ANOTHER GAME?"
 205 GET A$ : IF A$ = "Y" THEN POKE V + 27,0
       : PRINT "{CLR}" : GOTO 10
 210 IF A$ < > "N" THEN 205
 215 POKE V + 27,0 : POKE V + 21,0 : POKE 54273,0
       : POKE 54272,0 : POKE 54287,0 : POKE 54286,0
       : END
1000 POKE 2040,196 : T$ = TI$ : POKE 54273,34
       : POKE 54272,75 : POKE 54276,0 : POKE 54277,240
1005 POKE 54276,17 : POKE V + 28, PEEK ( V + 28 ) OR 8
       : POKE V + 6, PEEK ( V )
       : POKE V + 7, PEEK ( V + 1 )
1010 POKE V + 21, PEEK ( V + 21 ) OR 8 : POKE 2043,197
1015 FOR I = PEEK ( V + 1 ) TO 0 STEP - .1
       : POKE V + 7,I : NEXT
1020 POKE 2040,192 : CS = 0
       : POKE V + 21, PEEK ( V + 21 ) - 8
       : POKE V + 28, PEEK ( V + 28 ) - 8
       : POKE 2043,198
1025 CO = CO + 1 : GOTO 2030
2000 POKE 2041,195 : T$ = TI$ : POKE 54287,34
       : POKE 54286,75 : POKE 54290,0 : POKE 54291,240
2005 POKE 54290,33 : POKE V + 28, PEEK ( V + 28 ) OR 4
       : POKE V + 4, PEEK ( V + 2 )
       : POKE V + 5, PEEK ( V + 3 )
2010 POKE V + 21, PEEK ( V + 21 ) OR 4 : POKE 2042,197
2015 FOR I = PEEK ( V + 3 ) TO 0 STEP - .1
       : POKE V + 5,I : NEXT
2020 POKE 2041,192 : PS = 0
       : POKE V + 21, PEEK ( V + 21 ) - 4
       : POKE V + 28, PEEK ( V + 28 ) - 4
       : POKE 2042,198
2025 YO = YO + 1
2030 A = PEEK ( V + 30 ) : TI$ = T$ : GOTO 50
10000 DATA 0,4,0,0,21,0,0,12,0,0,12,0,2,170,160,10,170,
       168,8,170,136,8,42,8
10005 DATA 8,42,8,8,85,8,11,106,56,0,21,0,0,21,0,0,17,
       0,0,81,64,0,81,64,0,64,64
```

```
10010 DATA 0,64,64,0,64,64,0,64,64,0,64,64,99
10015 DATA 0,0,0,0,4,0,0,21,0,80,12,0,48,12,0,42,42,
        64,10,170,160,0,170,8
10020 DATA 0,42,8,0,42,8,0,21,32,0,41,48,0,21,0,0,21,0,
        0,81,0,1,65,0,1,1,0,1,1,0
10025 DATA 1,1,64,1,0,84,5,0,4,99
10030 DATA 0,0,0,0,16,0,0,84,0,0,48,5,0,48,12,2,168,
        168,10,170,160,32,170,0
10035 DATA 32,168,0,32,168,0,8,84,0,12,104,0,0,84,0,0,84,
        0,0,69,0,0,65,64,0,64
10040 DATA 64,0,64,64,1,64,64,21,0,64,16,0,80,99
10045 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
        0,0,0,0,0,0,0,0,0,0,0,0,0
10050 DATA 0,0,0,0,4,0,5,21,165,80,174,165,64,2,128,
        17,0,160,4,0,128,0,2,128
10055 DATA 0,58,0,0,99
10060 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
        0,0,0,0,0,0,0,0,0,0,0,0
10065 DATA 0,0,0,0,80,0,16,5,90,84,1,90,186,68,2,128,
        16,10,0,0,2,0,0,2,128
10070 DATA 0,0,172,99
10075 DATA 0,12,0,2,12,32,10,42,40,10,170,168,10,170,
        168,10,170,168,10,170,168
10080 DATA 10,187,168,10,170,168,10,170,168,10,42,40,
        10,42,40,2,42,32,2,34,32
10085 DATA 2,34,32,0,34,0,0,34,0,0,34,0,0,34,0,0,34,0,
        0,0,0,99
10090 DATA 0,0,0,0,0,0,0,0,0,0,16,0,
10095 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
        0,0,0,0,0,0,0,0,0,0,0,0,0,0
10100 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,99
10105 DATA 0,0,0,0,8,0,0,170,128,2,170,160,10,128,168,
        10,0,40,10,0,40,10,1,40
10110 DATA 10,5,104,10,3,40,10,5,104,10,1,40,10,1,40,
        10,1,40,5,85,84,0,4,0
10115 DATA 1,4,16,1,4,16,1,85,80,1,0,16,1,0,16
```



SPRITE

PLAYER/ENEMY
SHOOTING



SPRITE

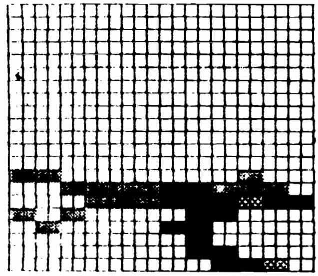
CARRIAGE

CHEXSUM

1=2925
5=2932
10=4111
15=4463
20=4271
25=4531
45=5503
50=5264
55=3327
57=2540
60=4343
65=3156
70=1156
75=1034
80=2053
85=5226
90=3329
100=1466
105=5274
110=3327
125=879
130=4644
135=576
150=869
155=4586
160=578
165=5046
170=2650
175=1797
176=828
177=1741
180=1725
182=518
185=1236
190=3663
195=2409
200=1390
205=2839
210=1121
215=4090

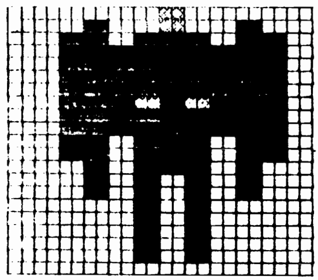
1000=3914
1005=4618
1010=2137
1015=2585
1020=5052
1025=1383
2000=3918
2005=4863
2010=2137
2015=2585
2020=5067
2025=748
2030=2074
10000=3837
10005=4259
10010=2033
10015=3561
10020=4341
10025=1322
10030=3815
10035=4178
10040=2195
10045=4315
10050=3905
10055=710
10060=4315
10065=3751
10070=656
10075=4170
10080=4060
10085=2939
10090=1408
10095=4315
10100=1993
10105=4012
10110=3698
10115=2053

TOTAL= 230344



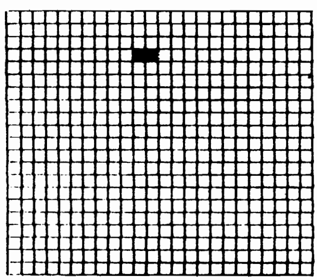
SPRITE

DEAD
PLAYER



SPRITE

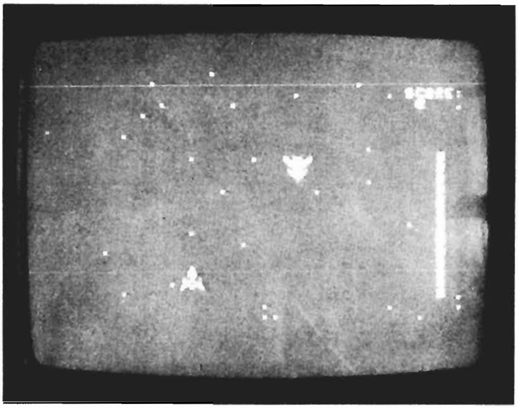
PLAYER/ENEMY
GHOST



SPRITE

BULLET

Earth Defense



You are left with insufficient fuel to make it back to earth so you embark on a suicide mission to kill as many aliens as possible with your remaining fuel.

Your controls are:

'W' up
'X' down
'A' left
'D' right
'S' to fire
'space bar' for shields

You have shields to protect your spaceship, however these use a lot of fuel and only last for a couple of seconds.

While shielded your craft will be a grey colour.

PROGRAM STRUCTURE	LINES
INITIALIZATION	0 — 4
SET UP DISPLAY	5 — 9
GET KEYBOARD CHARACTER	10
UPDATE SCORE	11
TEST WHICH KEY PRESSED	13 — 18
FIRE PLAYER'S MISSILES	20 — 23, 900 — 930
SHIELD CONTROL	24
UPDATE FUEL	25
RANDOM DIRECTION OF COMPUTER	26 — 30
COMPUTER FIRES MISSILE	31 — 39, 950 — 970
TEST IF PLAYER IS HIT	40
MISSILES COLLIDE	42 — 49
PLAYER EXPLODES	800 — 840
PLAYER HITS SHIP	21, 200 — 210
DATA	9000 — 9140

VARIABLES
PE = LAST KEYSTROKE
F = FUEL
SH = INCREMENT FOR PLAYER'S MISSILE
P = HORIZONTAL POSITION OF PLAYER
YP = VERTICAL POSITION OF PLAYER
SY = VERTICAL POSITION OF PLAYER'S MISSILE
SX = HORIZONTAL POSITION OF PLAYER'S MISSILE
X = HORIZONTAL POSITION OF COMPUTER
Y = VERTICAL POSITION OF COMPUTER
EX = HORIZONTAL POSITION OF COMPUTER'S MISSILE
EY = VERTICAL POSITION OF COMPUTER'S MISSILE

EARTH DEFENCE

```
0 POKE 55,255 : POKE 56,47 : PRINT "{CLR}"
    : POKE 53281,0 : POKE 53280,0 : V = 53248
    : CO = 54272
1 SM = 0 : FOR J = 1 TO 30
    : I = RND ( 1 ) * 1000 + 55296 : POKE I,1
    : POKE I - CO,46 : NEXT
2 POKE V + 21,31 : POKE 2040,192 : POKE 2041,193
    : POKE 2042,194 : POKE 2043,195 : POKE 2044,196
3 POKE V + 39,8 : POKE V + 40,3 : POKE V + 41,7
    : POKE V + 42,10 : POKE V + 43,5
4 FOR I = 0 TO 318 : READ Q : POKE 12288 + I,Q
    : NEXT : POKE 54296,1 : POKE 54278,240
    : POKE 54276,129
5 P = 120 : YP = 200 : POKE V,P : POKE V + 1,YP
    : X = 50 + INT ( RND ( 1 ) * 180 ) : Y = 0
    : POKE V + 2,X : POKE V + 3,Y
7 PE = PEEK ( V + 30 ) : F = 0 : POKE 54272,108
    : POKE 54273,223
8 PRINT "{HOME}{DOWN}{DOWN}{DOWN}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{WHT}SCORE:"
    : POKE V + 4,200
9 PRINT "{HOME}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{RIGHT}{RIGHT}{RIGHT}{WHT}{RVS ON}{SPACE}{DOWN}{LEFT}{SPACE}
    {DOWN}{LEFT}{SPACE}{DOWN}{LEFT}{SPACE}{DOWN}{LEFT}{SPACE}{DOWN}{LEFT}{SPACE}{DOWN}{LEFT}{SPACE}{DOWN}
    {LEFT}{SPACE}{DOWN}{LEFT}{SPACE}{DOWN}{LEFT}{SPACE}{DOWN}{LEFT}{SPACE}{DOWN}{LEFT}{SPACE}{DOWN}{LEFT}{SPACE}"
10 PE = PEEK ( 197 ) : F = F + .01 : POKE V + 39,8
    : IF HH < > 0 THEN POKE V + 39,12
11 PRINT "{HOME}{DOWN}{DOWN}{DOWN}{DOWN}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{RIGHT}{GRN}";S
12 POKE 55613 + INT ( F ) * 40,4
13 IF PE = 10 AND P > 50
    THEN P = P - 5 : F = F + .01
14 IF PE = 18 AND P < 239
    THEN P = P + 5 : F = F + .01
15 IF PE = 9 AND YP > 175 THEN YP = YP - 5
    : F = F + .01
16 IF PE = 23 AND YP < 230 THEN YP = YP + 5
    : F = F + .01
17 IF PE = 13 AND SH = 0 THEN GOSUB 900
18 IF PE = 60 AND HH = 0 THEN HH = 1 : F = F + 1
19 IF SH = 0 THEN 24
20 FOR I = 1 TO 2 : SY = SY - SH : POKE V + 5,SY
21 IF PEEK ( V + 30 ) = 6 THEN GOSUB 200
22 IF SY < 60 THEN SY = 0 : SH = 0 : SX = 150
    : POKE V + 5,0
23 NEXT
24 IF HH < > 0 THEN HH = HH + 1
    : IF HH = 10 THEN HH = 0
25 POKE V,P : POKE V + 1,YP : IF F > 15 THEN 100
26 IF RND ( 1 ) < .1
    THEN RN = INT ( RND ( 1 ) * 3 - 1 )
    : IF RN = 0 AND RND ( 1 ) < .1 THEN 26
27 IF X > 239 THEN RN = - 1
28 X = X + RN * 4 : IF X < 50 THEN RN = 1
29 Y = Y + 4 : IF Y > 235 THEN Y = 0
    : X = 50 + INT ( RND ( 1 ) * 180 )
    : SM = SM + 1
```

```
30 POKE V + 2,X : POKE V + 3,Y
31 IF EY = 0 AND RND ( 1 ) < .1 AND Y < 220
     THEN EY = Y + 20
     : EX = X : POKE V + 6,X : GOSUB 950
32 IF EY = 0 THEN GOTO 10
35 FOR I = 1 TO 2 : EY = EY + 6 : POKE V + 7,EY
     : POKE V + 6,EX
36 IF RND ( 1 ) < .2
     THEN EX = EX + ( ( EX > P ) - ( EX < P ) ) * 5
37 IF EX < 35 THEN EX = 35
38 IF EX > 238 THEN EX = 238
39 IF EY > 235 THEN EY = 0 : EX = 0 : POKE V + 7,EY
     : POKE V + 6,EX : GOTO 10
40 IF PEEK ( V + 30 ) = 9 AND HH = 0 THEN 300
42 NEXT : IF PEEK ( V + 30 ) < > 12 THEN 10
43 POKE 54273,0 : POKE 54272,0 : POKE 54296,5
     : FOR I = 30 TO 1 STEP - 1 : POKE 54273,I
     : NEXT
44 POKE 54296,1 : : POKE 54273,223 : POKE 54272,108
46 POKE V + 5,0 : POKE V + 7,0 : POKE V + 8,EX
     : POKE V + 9,EY
47 FOR I = 1 TO 100 : NEXT
48 EX = 50 : EY = 0 : SY = 0 : SX = 0 : SH = 0
     : POKE V + 8,0 : POKE V + 9,0
49 GOTO 10
100 GOSUB 800
110 PRINT "{CLR}{E}YOU'VE RUN OUT OF ENERGY!"
120 PRINT "{DOWN}{DOWN}{DOWN}{↑}HARD LUCK, WELL DONE 
     FOR SURVIVING FOR SO LONG!"
150 PRINT "{S}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}
     {▓}-------------------------------------"
160 PRINT "{DOWN}{DOWN}DO YOU WANT ANOTHER GAME?"
170 GET A$ : IF A$ = "Y" THEN RUN
180 IF A$ < > "N" THEN 170
190 PRINT "{◣}{DOWN}{DOWN}BYE"
195 END
200 POKE 54272,0 : POKE 54296,10
     : FOR I = 50 TO 1 STEP - 1.5 : POKE 54273,I
     : NEXT
201 POKE 54296,1 : POKE 54273,223 : POKE 54272,108
202 POKE V + 2,0 : POKE V + 3,0 : POKE V + 8,X
     : POKE V + 9,Y
205 S = S + 10 : IF Y = SY THEN S = S + 40
206 FOR Z = 1 TO 100 : NEXT : POKE V + 8,0
     : POKE V + 9,0
210 SH = 0 : SY = 0 : SX = 0 : Y = 0 : RETURN
300 GOSUB 800
310 PRINT "{CLR}YOU GOT HIT BY AN ENEMY MISSILE!!"
315 PRINT "{▓}{DOWN}{DOWN}HARD LINES!"
320 PRINT "{DOWN}{DOWN}{±}SCORE:";S
325 PRINT "ENERGY UNITS USED"; INT ( F )
330 PRINT "{↑}ENEMY CRAFT NOT DESTROYED";SM
340 PRINT "{DOWN}{DOWN}{←}NOT BAD!"
350 GOTO 150
800 FOR I = 1 TO 7 : POKE V + I,0 : NEXT
801 POKE 54273,0 : POKE 54272,0 : POKE 54276,0
```

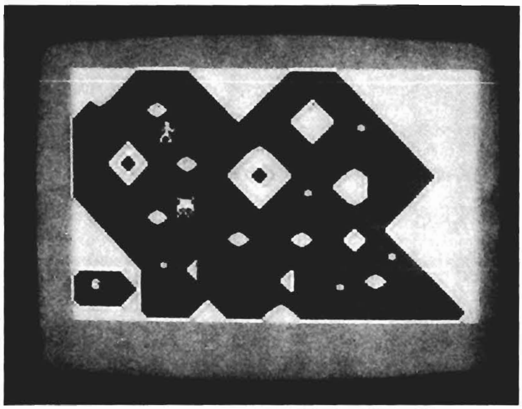
```
 805 POKE 54296,15 : POKE 54276,129
 810 POKE V + 8,P : POKE V + 9,YP
 820 FOR I = 50 TO 1 STEP - 1 : POKE 54273,I : NEXT
       : POKE 54273,0 : POKE 54272,0 : POKE 54296,1
 830 POKE V + 8,0 : POKE V + 9,0
 840 RETURN
 900 POKE 53280,2 : SH = 6 : SX = P : SY = YP
       : POKE V + 4,SX : F = F + .05 : POKE 53280,0
 910 POKE 54273,0 : POKE 54272,0 : POKE 54296,10
       : FOR I = 0 TO 150 STEP 3 : POKE 54273,I
       : NEXT
 920 POKE 54273,0 : POKE 54296,1
 930 RETURN
 950 POKE 54273,0 : POKE 54272,0 : POKE 54276,0
       : POKE 54276,33
 960 POKE 54273,50 : FOR ZX = 1 TO 50 : NEXT
       : POKE 54276,129 : POKE 54273,233
       : POKE 54272,108
 970 RETURN
9000 DATA 0,0,0,0,8,0,0,28,0,0,28,0,0,62,0,0,127,0,
       0,127,0,0,127,0,0,127,0,2
9010 DATA 127,32,2,28,32,7,28,112,2,62,32,6,255,
       176,7,235,240,7,235,240,15,255
9020 DATA 248,15,62,120,28,28,60,24,0,12,0,0,0,99
9030 DATA 0,130,0,64,130,4,112,214,28,120,254,60,
       124,124,124,62,56,248,31,255
9040 DATA 240,15,255,224,7,255,192,1,255,0,0,254,0,
       12,108,96,6,108,192,7,255,192
9050 DATA 3,255,128,1,255,0,0,254,0,0,56,0,0,16,0,0,
       16,0,0,16,0,99
9060 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
       0,0,0,0,0,0,0,0,0,0,0,0,2,0
9070 DATA 32,7,0,112,15,128,248,31,193,252,26,
       193,172,18,65,36,2,0,32,0,0,0
9080 DATA 0,0,0,2,0,32,99
9090 DATA 0,0,0,0,0,0,0,0,0,0,1,215,0,0,254,0,0,
       254,0,0,124,0,0,16,0,0
9100 DATA 16,0,0,16,0,0,124,0,0,56,0,0,16,0,0,16,
       0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
9110 DATA 0,0,0,0,0,0,99
9120 DATA 3,12,0,0,3,0,192,192,195,3,204,192,12,3,
       12,0,12,48,0,51,3,48,51,0
9130 DATA 3,12,0,0,3,0,192,192,195,3,204,192,12,
       3,12,0,12,48,0,51,3,48,51,0
9140 DATA 3,12,0,0,3,0,192,192,195,3,204,195,3,204,195,3,204
```

CHEXSUM

0=4078
1=4460
2=4072
3=3594
4=4337
5=5883
7=2812
8=2574
9=7217
10=3993
11=1306
12=1343
13=2724
14=2782
15=3010
16=3053
17=1656
18=2374
19=814
20=2395
21=1541
22=2931
23=131
24=2886
25=2034
26=4158
27=1334
28=2116
29=4305
30=1284
31=4612
32=1197
35=3074
36=3333
37=1233
38=1371
39=3900
40=1950
42=1763
43=3958
44=2081
46=2810
47=964
48=3733
49=472
100=302
110=2402
120=4618
150=3033
160=2289
170=1263
180=1125
190=673
195=129
200=3518
201=2025
202=2616
205=2122
206=2345
210=1968
300=302
310=2994
315=1255
320=926
325=2034
330=2571
340=864
350=526
800=1528
801=1772
805=1337
810=1376
820=3898
830=1218
840=143
900=4412
910=3900
920=1167
930=143
950=2438
960=3985
970=143
9000=3986
9010=4328
9020=2290
9030=4196
9040=4557
9050=3223
9060=4317
9070=3964
9080=926
9090=3439
9100=4456
9110=863
9120=3912
9130=3912
9140=2875

TOTAL= 244252

Jelly Maze



How may points can you get before the jelly creature kills you?

There are 5 Power Pills in the maze, eat one of these and you will turn blue. When you are blue you can eat the jelly creature, but be careful you will not remain blue for longer than a few seconds.

You gain 1 point for moving, 100 points for eating a Power Pill and 500 points for eating a jelly creature. If you eat one, another will apear.

Your controls are:

- 'W' up
- 'X' down
- 'A' left
- 'D' right

PROGRAM STRUCTURE	**LINES**
INITIALIZATION	0 — 8
DISPLAY SET-UP	9 — 49
GET KEYBOARD CHARACTER	51
DISPLAY PRESENT SCORE	51
UPDATE POSITION OF PLAYER	52 — 70
TEST FOR PLAYER COLLISION	71
TAKE ACTION ON COLLISION	72 — 85
DISPLAY PLAYER AT NEW POSITION	86 — 93
MOVE JELLY	95 — 150
PLAYER EATS JELLY	1000 — 1040
JELLY EATS PLAYER	2000 — 2010
PLAYER EATS POWER PILL	5000 — 5010
DATA	9000 — 9090

VARIABLES

SC = SCREEN REFERENCE POINT
X1 = HORIZONTAL POSITION OF PLAYER
Y1 = VERTICAL POSITION OF PLAYER
DM = ACTUAL POSITION OF PLAYER
DC = ACTUAL POSITION OF COMPUTER RELATIVE TO SC
D = DIRECTION OF PLAYER
S = PLAYER'S SCORE

JELLY MAZE

```
0 POKE 55,255 : POKE 56,47 : PRINT "♥"
    : POKE 53280,2 : POKE 53281,0
1 V = 53248 : POKE V + 22,216 : POKE V + 24,28
2 POKE V + 34,7 : POKE V + 35,5
3 FOR I = 12288 TO 12471 : READ Q : POKE I,Q
    : NEXT
4 FOR I = 12544 TO 12551 : POKE I,0 : NEXT
5 FOR I = 12672 TO 12751 : READ Q : POKE I,Q : NEXT
6 POKE 54296,10 : POKE 54276,0 : POKE 54276,33
    : POKE 54278,240
7 POKE 54273,0 : POKE 54272,0
8 SC = 1024 : CS = 55296 : X = 4 : Y = 4 : D = 1
    : A = 15 : B = 20 : AD = 1 : DI = 3
9 PRINT "SIIIIIIGNNNNNFIIIIIIIIIGNNNNFIIIIIIIIIIIII";
10 PRINT "IIIIIG▲ ▲ ▲ ▲ ▲ ▲ ▲FIIIIIIG▲ ▲ ▲ ▲ ▲
    ▲FIIIIIIIIIIII";
12 PRINT "IIIIG▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲FIIIIG▲ ▲ ▲ ▲ ▲
    ▲ ▲ ▲FIIIIIIIIIII";
14 PRINT "IGLM▲ ▲ ▲ ▲JK▲ ▲ ▲ ▲ ▲FIIG▲ ▲ ▲ ▲HE▲
    ▲ ▲ ▲FIIIIIIIIIII";
15 PRINT "G▲ ▲ ▲ ▲ ▲ ▲ ▲LM▲ ▲ ▲ ▲ ▲ ▲FG▲ ▲ ▲ ▲
   HIIE▲ ▲ ▲ ▲FIIIIIIIII";
18 PRINT "P▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲
    ▲ ▲FIIG▲ ▲D▲ ▲FIIIIIIII";
20 PRINT "P▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲
    ▲ ▲ ▲FG▲ ▲ ▲ ▲ ▲ ▲ ▲FIIIIIII";
22 PRINT "P▲ ▲ ▲ ▲HE▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲JK▲ ▲
    ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲FIIIIII";
24 PRINT "P▲ ▲ ▲HGFE▲ ▲ ▲JK▲ ▲ ▲ ▲HIIE▲ ▲ ▲
    ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ FIIIII";
25 PRINT "P▲ ▲ ▲FEHG▲ ▲ ▲LM▲ ▲ ▲HIGFIE▲ ▲ ▲ ▲
    ▲JK▲ ▲ ▲ ▲ ▲ ▲FIIII";
26 PRINT "P▲ ▲ ▲ ▲FG▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲FIEHIG▲ ▲
    ▲ ▲HIIP▲ ▲ ▲ ▲ ▲HIIII";
28 PRINT "P▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲FIIG▲
    ▲D▲ ▲FIIP▲ ▲ ▲ ▲HIIIII";
30 PRINT "E▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲FG▲
    ▲ ▲ ▲ ▲ ▲ ▲LM▲ ▲ ▲ ▲HIIIIII";
32 PRINT "IE▲ ▲ ▲ ▲ ▲ ▲JK▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲
    ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲HIIIIIII";
34 PRINT "IIE▲ ▲ ▲ ▲ ▲LM▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲
    ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲HIIIIIIII";
36 PRINT "IIIE▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲JK▲ ▲ ▲ ▲JK
    ▲ ▲ ▲HE▲ ▲FIIIIIIII";
38 PRINT "IIIIE▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲LM▲ ▲ ▲ ▲LM
    ▲ ▲ ▲FG▲ ▲ ▲FIIIIIII";
39 PRINT "IIIIIE▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲
    ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲D▲FIIIIII";
40 PRINT "IIIIIIE▲ ▲D▲ ▲HE▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲
    ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲FIIIII";
42 PRINT "GNNNNFIP▲ ▲ ▲ ▲FG▲ ▲ ▲ ▲ ▲ ▲ ▲HP▲ ▲ ▲
    ▲ ▲ ▲JK▲ ▲ ▲ ▲FIIII";
43 PRINT "P▲ ▲ ▲ ▲ ▲FP▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲FP
    ▲ ▲ ▲D▲ ▲LM▲ ▲ ▲ ▲ ▲FIII";
```

```
44 PRINT "P▲▲▲▲▲HP▲▲▲▲▲▲▲▲▲▲▲▲▲
      ▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲FII";
45 PRINT "EOOOOHIP▲▲▲▲▲HE▲▲▲▲▲JK▲▲▲▲
      ▲▲▲▲▲▲▲▲▲▲▲▲FI";
46 PRINT "IIIIIIIEOOOOHIIEOOOHIIEOOOOOOOOOOOOOOOHI";
47 FOR I = 960 TO 999 : POKE CS + I,2
      : POKE SC + I,9 : NEXT : M = SC + X + Y * 40
      : MC = CS + X + Y * 40
48 POKE MC,4 : POKE MC + 1,4 : POKE MC + 40,4
      : POKE MC + 41,4
49 POKE M,17 : POKE M + 1,18 : POKE M + 40,19
      : POKE M + 41,20
50 FOR Z = 1 TO DI
51 PE = PEEK ( 197 ) : PRINT "[S][Q][Q][Q][Q][Q][Q][Q][Q][Q][Q]
      [Q][Q][Q][Q][Q][Q][Q][Q][Q][Q][Q][D][E]"S
52 M = SC + X + Y * 40 : X1 = X : Y1 = Y
55 IF PE < > 9 AND PE < > 23 AND PE < > 10
      AND PE < > 18 THEN 95
58 IF PE = 9 THEN Y1 = Y1 - 1
60 IF PE = 23 THEN Y1 = Y1 + 1
65 IF PE = 10 THEN X1 = X1 - 1
66 WHER$ = "MAN"
70 IF PE = 18 THEN X1 = X1 + 1
71 DM = SC + X1 + Y1 * 40 : P1 = PEEK ( DM )
      : P2 = PEEK ( DM + 1 ) : P3 = PEEK ( DM + 40 )
      : P4 = PEEK ( DM + 41 )
72 IF P1 = 4 THEN GOTO 5000
73 IF P2 = 4 THEN GOTO 5000
74 IF P3 = 4 THEN GOTO 5000
75 IF P4 = 4 THEN GOTO 5000
76 IF P1 < 4 THEN 1000
77 IF P2 < 4 THEN 1000
78 IF P3 < 4 THEN 1000
79 IF P4 < 4 THEN 1000
82 IF P1 < 17 THEN POKE 54276,0 : POKE 54276,33
      : POKE 54273,20 : GOTO 95
83 IF P2 < 17 THEN POKE 54276,0 : POKE 54276,33
      : POKE 54273,20 : GOTO 95
84 IF P3 < 17 THEN POKE 54276,0 : POKE 54276,33
      : POKE 54273,20 : GOTO 95
85 IF P4 < 17 THEN POKE 54276,0 : POKE 54276,33
      : POKE 54273,20 : GOTO 95
86 DC = CS + X1 + Y1 * 40 : POKE M,32
      : POKE M + 1,32 : POKE M + 40,32
      : POKE M + 41,32
87 IF AD = 1 THEN POKE DC,4 : POKE DC + 1,4
      : POKE DC + 40,4 : POKE DC + 41,4
88 IF AD = - 1 THEN POKE DC,3 : POKE DC + 1,3
      : POKE DC + 40,3 : POKE DC + 41,3
89 S = S + 1
90 POKE DM,17 : POKE DM + 1,18
91 IF D = 1 THEN POKE DM + 40,19 : POKE DM + 41,20
92 IF D = - 1 THEN POKE DM + 40,21 : POKE DM + 41,22
93 X = X1 : Y = Y1 : D = - D
95 POKE 54273,0 : NEXT : A1 = A : B1 = B
      : M = SC + A + B * 40
```

```
 96 IF RND ( 1 ) < .8 THEN A1 = A1 + ( ( A1 > X )
       - ( A1 < X ) ) * AD
 98 DM = SC + A1 + B1 * 40 : P1 = PEEK ( DM )
       : P2 = PEEK ( DM + 1 ) : P3 = PEEK ( DM + 40 )
       : P4 = PEEK ( DM + 41 )
 99 IF ( P1 = 32 OR P1 < 5 ) AND ( P2 = 32 OR P2 < 5 )
       AND ( P3 = 32 OR P3 < 5 )
       AND ( P4 = 32 OR P4 < 5 ) THEN A = A1
 100 IF A1 < > A AND RND ( 1 ) < .5 THEN A1 = A : A1 = A1 +
       INT ( RND ( 1 ) * 3 ) - 1 : GOTO 98
 101 WHER$ = "JELLY"
 104 IF RND ( 1 ) < .8 THEN B1 = B1 + ( ( B1 > Y )
       - ( B1 < Y ) ) * AD
 105 DM = SC + A + B1 * 40 : P1 = PEEK ( DM )
       : P2 = PEEK ( DM + 1 ) : P3 = PEEK ( DM + 40 )
       : P4 = PEEK ( DM + 41 )
 106 IF P1 > 16 AND P1 < 32 THEN 1000
 107 IF P2 > 16 AND P2 < 32 THEN 1000
 108 IF P3 > 16 AND P3 < 32 THEN 1000
 109 IF P4 > 16 AND P4 < 32 THEN 1000
 110 IF P1 > 255 OR P2 > 255 OR P3 > 255 OR P4 > 255
       THEN 1000
 111 IF ( P1 = 32 OR P1 < 5 )
       AND ( P2 = 32 OR P2 < 5 ) AND
       ( P3 = 32 OR P3 < 5 ) AND
       ( P4 = 32 OR P4 < 5 ) THEN B = B1
 115 IF B1 < > B THEN B1 = B : B1 = B1 +
       INT ( RND ( 1 ) * 3 ) - 1 : GOTO 105
 120 DM = SC + A + B * 40 : DC = CS + A + B * 40
 130 POKE DC,14 : POKE DC + 1,14 : POKE DC + 40,14
       : POKE DC + 41,14
 135 POKE M,32 : POKE M + 1,32 : POKE M + 40,32
       : POKE M + 41,32
 140 POKE DM,0 : POKE DM + 1,1 : POKE DM + 40,2
       : POKE DM + 41,3
 145 IF NU < > 0 THEN NU = NU + 1
       : IF NU = 20 THEN NU = 0 : AD = 1
 146 IF RND ( 1 ) < .03 AND DI > 1 THEN DI = DI - 1
 147 IF B = B1 OR A = A1 THEN POKE 54276,0
       : POKE 54276,33 : FOR I = 0 TO 5 STEP .3
       : POKE 54273,I : NEXT
 148 POKE 54273,0
 150 GOTO 50
1000 IF AD = 1 THEN 2000
1001 POKE 54276,0 : POKE 54276,33 : FOR I = 0 TO 150
       STEP 4 : FOR J = 1 TO 20 : NEXT
1002 POKE 54273,I : NEXT : POKE 54273,0
1004 M = SC + X + Y * 40 : DM = SC + X1 + Y1 * 40
1005 POKE M,32 : POKE M + 1,32 : POKE M + 40,32
       : POKE M + 41,32
1006 POKE DM,32 : POKE DM + 1,32 : POKE DM + 40,32
       : POKE DM + 41,32
1010 M = SC + A + B * 40 : DM = SC + A1 + B1 * 40
1015 POKE M,32 : POKE M + 1,32 : POKE M + 40,32
       : POKE M + 41,32
1016 POKE DM,32 : POKE DM + 1,32 : POKE DM + 40,32
```

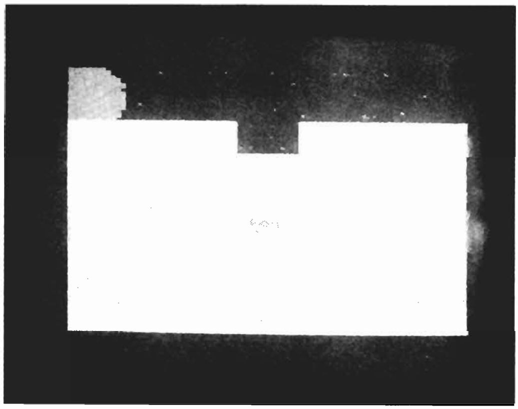
```
     : POKE DM + 41,32
1020 X = 4 : Y = 4 : A = 15 : B = 20
1035 S = S + 500 : AD = 1
1040 GOTO 50
2000 POKE 54276,0 : POKE 54276,129 : POKE 54273,10
2001 FOR I = 1 TO 700 : NEXT
2002 FOR I = 240 TO 0 STEP - 2 : POKE 54273,I : NEXT
2003 POKE 54273,0
2005 POKE V + 24,20 : PRINT "■ ■HARD ▲LINES,THE
     ▲JELLY ▲CAUGHT ▲YOU!"
2010 PRINT "■ ■ ■YOU ▲SCORED"S"POINTS. ■"
2020 IF S < 200 THEN PRINT "◣HA,HA!"
2030 IF S > 199 AND S < 400 THEN PRINT "■NOT ▲BAD."
2040 IF S > 399 AND S < 800 THEN PRINT "■SUPER!"
2050 IF S > 799 THEN PRINT "■AMAZING!"
2060 PRINT "■ ■ ■ANOTHER ▲GO?"
2070 GET A$ : IF A$ = "Y" THEN RUN
2080 IF A$ < > "N" THEN 2070
2090 PRINT "◣BYE!"
2100 END
5000 S = S + 100 : AD = - 1 : NU = 1
5002 POKE 54276,0 : POKE 54276,17 : FOR I = 10 TO 50
     : POKE 54273,I : NEXT
5005 POKE 54273,0
5010 GOTO 86
9000 DATA 32,131,47,55,63,63,255,254,8,8,224,
     112,240,240,188,172
9010 DATA 162,130,136,130,34,34,128,128,44,8,8,
     32,32,8,131,128
9015 DATA 0,0,4,29,63,29,4,0
9016 DATA 64,64,208,208,244,244,253,253,
     127,127,31,31,7,7,1,1
9017 DATA 253,253,244,244,208,208,64,64,1,
     1,7,7,31,31,127,127
9018 DATA 255,255,255,255,255,255,255,255,0,
     0,0,1,7,31,31,127
9019 DATA 0,0,0,64,208,244,244,253
9020 DATA 127,31,31,7,1,0,0,0,253,244,244,208,64,0,0,0
9022 DATA 85,85,0,0,0,0,0,0,0,0,0,0,0,0,85,85
9024 DATA 64,64,64,64,64,64,64,64
9025 DATA 1,3,3,1,1,7,13,25,128,192,192,
     128,128,226,182,156
9030 DATA 25,49,3,6,12,12,6,30,128,128,192,96,
     56,24,48,0
9040 DATA 1,3,7,12,24,48,24,48,128,128,192,
     192,96,48,48,240
9050 DATA 0,60,102,102,102,102,102,60,0,48,
     112,48,48,48,48,124
9060 DATA 0,60,102,6,12,48,96,126,0,60,
     102,6,28,6,102,60
9070 DATA 0,6,14,30,102,127,6,6,0,126,96,
     120,6,6,102,60
9080 DATA 0,60,102,96,124,102,102,60,0,126,
     102,6,12,24,24,24
9090 DATA 0,60,102,102,60,102,102,60,0,60,
     102,102,62,6,102,60
```

CHEXSUM

```
0=2660
1=2139
2=1332
3=2103
4=1766
5=2098
6=2612
7=1167
8=4383
9=4017
10=3513
12=3354
14=3334
15=3126
18=2963
20=2793
22=2797
24=2945
25=2989
26=2988
28=2942
30=2771
32=2817
34=2931
36=3113
38=3089
39=2905
40=2898
42=3052
43=2767
44=2448
45=2721
46=4002
47=5707
48=2737
49=2667
50=765
51=1889
52=2209
55=3543
58=1427
60=1473
65=1469
66=847
70=1477
71=5999
72=1291
73=1291
74=1291
75=1295
76=901
77=901
78=901
79=905
82=3221
83=3221
84=3221
85=3225
86=4186
87=3423
88=3638
89=571
90=1233
91=2226
92=2418
93=1478
95=3034
96=3338
98=5958
99=5863
100=4472
101=1024
104=3341
105=5932
106=1586
107=1588
108=1588
109=1594
110=3090
111=5863
115=3731
120=2693
130=2929
135=2667
140=2746
145=3456
146=2347
147=4533
148=528
150=476
1000=905
1001=3434
1002=1365
1004=2789
1005=2667
1006=2953
1010=2734
1015=2667
1016=2953
1020=1582
1035=1137
1040=476
2000=1963
2001=964
2002=2038
2003=528
2005=4042
2010=1913
2020=1538
2030=2460
2040=2284
2050=1658
2060=1144
2070=1263
2080=1184
2090=661
2100=129
5000=1768
5002=3014
5005=528
5010=482
9000=3131
9010=3010
9015=1109
9016=2958
9017=2954
9018=2938
9019=1444
9020=2524
9022=2074
9024=1418
9025=2852
9030=2690
9040=2849
9050=2990
9060=2686
9070=2591
9080=2881
9090=2922

TOTAL= 342202
```

Dark Star



Test your skills against the enemy and the clock!

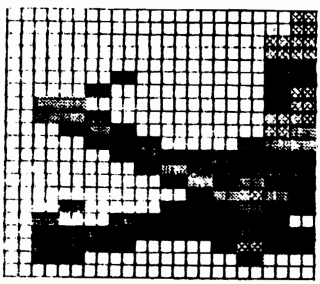
The Death Star will obliterate your home planet, when it comes into range from behind its moon, using the immense power of its space gun. Before this happens you will attempt to take as many attacking enemy fighters with you before you are shot down or they shoot your home planet.

You fly your X wing fighter along the trench on the Death Star engaging the enemy in mortal combat, using the 'Z' and 'C' keys to move left and right and 'M' to fire.

Straight Shooting!

PROGRAM STRUCTURE	LINES
INITIALIZATION	0 — 140
ROTATE COLOURS, READ KEY	150
MOVE PLAYER'S SHIP (TWO SPRITES)	155 — 167
MOVE PLANET	170 — 175
CREATE NEW ALIEN SHIP	180 — 185
MOVE ALIEN SHIP	190 — 215
ALIEN SHIP FIRES	217 — 240
PLAYER FIRES	250
SETUP SHOT SPRITE	251 — 260
KEEP MOVING SHOT, CHECKING IF IT HITS ALIEN SHIP	265
TURN OFF SHOT SPRITE, IF HIT INCREMENTS SCORE AND BLOW UP ALIEN ON SCREEN	270 — 297
TIME TO BLOW UP PLANET? IF SO THEN BLOW IT UP AND END GAME	300 — 325
GO TO START (150)	350
ROTATE COLOURS, (ANIMATE DEATH STAR)	500 — 505
GAME OVER, PRINT SCORE, ASK IF WANT NEW GAME	1000 — 1035
DATA	10000 — 20105

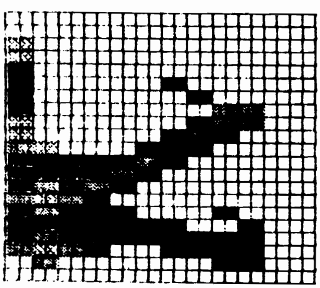
VARIABLES
V = VIDEO CHIP
CO = DIFFERENCE BETWEEN VIDEO AND SCREEN MEMORY
I, A, J = TEMPORARIES
B, C, M1, M2 = COLOUR VALUES
KB = LOCATION TO GET KEY PRESSED FROM
PL = PLANET POSITION
SC = SCORE
K = KEY PRESSED
X = POSITION OF PLAYER' SHIP
FT = FLAG TO INDICATE IF SHOOTING OR NOT
P = VALUE OF SPRITE HIGH POSITION REGISTER
X1 = POSITION OF ALIEN SHIP
DE = VALUE OF COLLISION REGISTER



SPRITE

LEFT SIDE OF PLAYER'S SHIP

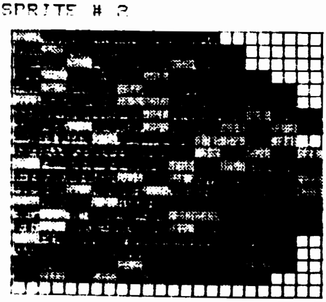
SPRITE # 1



SPRITE

RIGHT SIDE OF PLAYER'S SHIP

SPRITE # 2



SPRITE

MOON

DARK STAR

```
  0 POKE 55,255 : POKE 56,47 : V = 53248
      : POKE V + 32,0 : POKE V + 33,5 : CO = 54272
      : POKE 54296,15
  1 POKE 54278,128 : POKE 54277,240 : POKE 54276,0
      : POKE 54276,129
  2 POKE V + 34,6 : POKE V + 35,5 : POKE 2040,208
      : POKE 2041,209 : POKE 2044,210 : POKE V + 43,11
  3 POKE V + 2045,212 : POKE 54273,4 : POKE 54272,73
      : POKE 2045,213
  5 FOR I = 12544 TO 12551 : POKE I,0 : NEXT
      : FOR I = 12288 TO 12367 : READ A : POKE I,A
      : NEXT
  6 POKE V + 22, PEEK ( V + 22 ) OR 16
      : POKE V + 24,28 : POKE V + 39,14
      : POKE V + 40,14
  8 FOR I = 13312 TO 13759 : READ A : POKE I,A
      : NEXT : POKE V + 37,2 : POKE V + 38,8
  9 POKE V + 28,31 : POKE V,160 : POKE V + 1,220
      : POKE V + 2,184 : POKE V + 3,220
 10 PRINT "♥ ◙ ◙ ◙ ◙ ◙BBBBBBBBBBBBBBBBE@I@I@@HBBBB
     BBBBBBBBBBBB";
 15 PRINT "AAAAAAAAAAAAAAADB@@@I@@BGAAAAAAAAAAAAAAA";
 20 PRINT "AAAAAAAAAAAAAAADABII@@I@BAGAAAAAAAAAAAAAAA";
 25 PRINT " ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲CAABBBBBBBAAF
      ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲";
 30 PRINT " ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲C ▲AAAAAAAAAAA
      ▲F ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲";
 35 PRINT "BBBBBBBBBBBE ▲ ▲AAAAAAAAAAA ▲ ▲HBBBBBBB
     BBBB";
 40 PRINT "BBBBBBBBBBEB ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲
      ▲ ▲BHBBBBBBBBBB";
 45 PRINT "BBBBBBBBBEBB ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲
      ▲ ▲BBHBBBBBBBBB";
 50 PRINT "AAAAAAAADBBB ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲
      ▲ ▲BBBGAAAAAAAA";
 55 PRINT "AAAAAAADABBBBBBBBBBBBBBBBBBBBBBBAGAAAAAAA";
 60 PRINT "AAAAAADAABBBBBBBBBBBBBBBBBBBBBBAAGAAAAAA";
 65 PRINT "AAAAADAAABBBBBBBBBBBBBBBBBBBBBBAAAGAAAAA";
 70 PRINT " ▲ ▲ ▲ ▲CAAAABBBBBBBBBBBBBBBBBBBBBBBAAAAF
      ▲ ▲ ▲ ▲";
 75 PRINT " ▲ ▲ ▲C ▲AAAAAAAAAAAAAAAAAAAAAAAAAAAA
      ▲F ▲ ▲ ▲";
 80 PRINT " ▲ ▲C ▲ ▲AAAAAAAAAAAAAAAAAAAAAAAAAAAA
      ▲ ▲F ▲ ▲";
 85 PRINT " ▲C ▲ ▲ ▲AAAAAAAAAAAAAAAAAAAAAAAAAAAA
      ▲ ▲ ▲F ▲";
 90 PRINT "E ▲ ▲ ▲ ▲AAAAAAAAAAAAAAAAAAAAAAAAAAAA
      ▲ ▲ ▲ ▲H";
 95 PRINT "B ▲ ▲ ▲ ▲AAAAAAAAAAAAAAAAAAAAAAAAAAAA
      ▲ ▲ ▲ ▲B";
100 PRINT "B ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲
      ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲B";
105 PRINT "B ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲
      ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲";
```

```
110 POKE 2023 + CO,8 : POKE 2023,2
115 FOR I = 1024 TO 1223 : POKE I + CO,8 : POKE I,0
     : IF RND ( 1 ) > .9 THEN POKE I,9
120 NEXT : POKE 2042,211 : POKE V + 41,11
     : POKE V + 23,36 : POKE V + 29,4
     : POKE V + 4,24 : POKE V + 5,50
121 FOR I = 1224 TO 2047 : POKE CO + I,8 : NEXT
125 POKE V + 42,9 : POKE V + 6,0 : POKE V + 7,60
     : POKE 2043,212
130 POKE V + 21,15
140 B = 5 : M1 = 6 : M2 = 5 : KB = 197 : X = 160
     : PL = 0 : FI = 0 : SC = 0
150 GOSUB 500 : K = PEEK ( KB )
155 X = X + ( K = 12 ) * 4 - ( K = 20 ) * 4
     : IF X < 64 THEN X = 64
157 IF X > 260 THEN X = 260
160 P = PEEK ( V + 16 )
     : IF X > 255 THEN POKE V,X - 256
     : POKE V + 16,P OR 1 : GOTO 165
162 POKE V,X : POKE V + 16,P AND 254
165 P = PEEK ( V + 16 ) : X = X + 24
     : IF X > 255 THEN POKE V + 2,X - 256
     : POKE V + 16,P OR 2 : GOTO 170
167 POKE V + 2,X : POKE V + 16,P AND 253
170 J = 75 : IF FI THEN J = 1
172 X = X - 24 : FOR I = 1 TO J : NEXT
     : PL = PL + .5 : IF PL > 80 THEN PL = 80
175 POKE V + 6,PL : IF FI > 0 THEN 190
180 IF RND ( 1 ) < .9 THEN 250
185 POKE V + 8,175 : POKE V + 9,90
     : POKE V + 21, PEEK ( V + 21 ) OR 16 : FI = 1
     : X1 = PEEK ( V + 30 )
190 POKE V + 9, PEEK ( V + 9 ) + 4
     : X1 = PEEK ( V + 8 )
     : IF PEEK ( V + 16 ) AND 16 THEN X1 = X1 + 256
195 IF PEEK ( V + 9 ) > 234
     THEN POKE V + 21, PEEK ( V + 21 ) AND 239
     : FI = 0 : GOTO 250
200 X1 = X1 - 12 : X1 = X1 + ( X1 > X ) * 2
     - ( X1 < X ) * 2 : X1 = X1 + 12
205 IF X1 > 255 THEN POKE V + 8,X1 - 256
     : POKE V + 16, PEEK ( V + 16 ) OR 16 :GOTO 210
210 POKE V + 8,X1
     : POKE V + 16, PEEK ( V + 16 ) AND 239
215 IF PEEK ( V + 30 ) AND 16 THEN 1000
217 IF RND ( 1 ) < .92 THEN 250
218 POKE 54290,0 : POKE 54291,12 : POKE 54287,17
     : POKE 54286,37 : POKE 54292,0 : POKE 54290,17
220 IF X1 > 255 THEN POKE V + 10,X1 - 256
     : POKE V + 16, PEEK ( V + 16 ) OR 32 : GOTO 225
222 POKE V + 10,X1
     : POKE V + 16, PEEK ( V + 16 ) AND 223
225 POKE V + 11, PEEK ( V + 9 ) + 21 : POKE V + 44,8
     : POKE V + 21, PEEK ( V + 21 ) OR 32
     : DE = PEEK ( V + 30 )
230 DE = 0 : FOR J = PEEK ( V + 11 ) TO 250 STEP 4
```

```
      : POKE V + 11,J
      : IF PEEK ( V + 30 ) AND 32 THEN DE = 1
      : J = 250
235 NEXT J : IF DE THEN 1000
240 POKE V + 21, PEEK ( V + 21 ) AND 223
250 IF K < > 36 THEN 300
251 POKE 54290,0 : POKE 54291,12 : POKE 54287,137
      : POKE 54286,43 : POKE 54292,0 : POKE 54290,33
255 X = X + 12 : IF X > 255 THEN POKE V + 10,X - 256
      : POKE V + 16, PEEK ( V + 16 ) OR 32 : GOTO 260
257 POKE V + 10,X
      : POKE V + 16, PEEK ( V + 16 ) AND 223
260 POKE V + 11, PEEK ( V + 1 ) - 42 : POKE V + 44,4
      : POKE V + 21, PEEK ( V + 21 ) OR 32
      : DE = PEEK ( V + 30 )
265 DE = 0 : FOR J = PEEK ( V + 11 ) TO 70 STEP - 4
      : POKE V + 11,J
      : IF PEEK ( V + 30 ) AND 32 THEN DE = 1
      : J = 70
270 NEXT J : X = X - 12
      : POKE V + 21, PEEK ( V + 21 ) AND 223
      : IF DE = 0 THEN 300
290 SC = SC + 10 : FI = 0
      : POKE V + 21, PEEK ( V + 21 ) AND 239
291 POKE 54290,0 : POKE 54291,133 : POKE 54287,8
      : POKE 54286,147 : POKE 54290,129
295 POKE 2045,214 : POKE V + 44,2
      : POKE V + 21, PEEK ( V + 21 ) OR 32
      : FOR I = 1 TO 100 : NEXT
297 POKE 2045,213
      : POKE V + 21, PEEK ( V + 21 ) AND 223
300 IF PL < 80 OR RND ( 1 ) < .9 THEN 350
301 POKE 54290,0 : POKE 54291,12 : POKE 54287,137
      : POKE 54286,43 : POKE 54292,0 : POKE 54290,33
305 POKE V + 44,1 : POKE V + 10,PL : POKE V + 11,69
      : POKE V + 23, PEEK ( V + 23 ) AND 223
310 POKE V + 21, PEEK ( V + 21 ) OR 32
      : FOR I = 1 TO 500 : NEXT
311 POKE 54290,0 : POKE 54291,207 : POKE 54287,8
      : POKE 54286,147 : POKE 54290,129
312 POKE V + 21, PEEK ( V + 21 ) AND 247
315 POKE V + 44,10 : POKE V + 11, PEEK ( V + 7 )
      : POKE 2045,214
      : POKE V + 23, PEEK ( V + 23 ) OR 32
320 POKE V + 29, PEEK ( V + 29 ) OR 32
      : POKE V + 32, PEEK ( V + 33 )
      : FOR I = 1 TO 4000 : NEXT : POKE V + 32,0
325 POKE V + 29, PEEK ( V + 29 ) AND 223
      : GOTO 1007
350 GOTO 150
500 C = M2 : M2 = M1 : M1 = B : B = C
505 POKE V + 33,B : POKE V + 34,M1 : POKE V + 35,M2
      : RETURN
1000 POKE V + 21, PEEK ( V + 21 ) AND 252
      : POKE V + 10, PEEK ( V ) + 12
      : POKE V + 11, PEEK ( V + 1 )
```

```
1002 POKE V. + 32,1
1003 POKE 54290,0 : POKE 54291,207 : POKE 54287,8
      : POKE 54286,147 : POKE 54290,129
1005 POKE 2045,214 : POKE V + 44,2
      : POKE V + 21, PEEK ( V + 21 ) OR 32
      : FOR I = 1 TO 4000 : NEXT
1007 POKE V + 32,0 : POKE V + 33,6
1010 POKE 2045,213 : POKE V + 21,0
1015 POKE V + 24,20 : PRINT "[□][■]YOU SCORED";SC
1020 PRINT "[◘][◘][◘][◘][◘][◘]ANOTHER GAME?"
1022 POKE 54276,0 : POKE 54290,0 : POKE 54273,0
      : POKE 54272,0 : POKE 54287,0 : POKE 54286,0
1025 GET A$ : IF A$ < > "Y" THEN 1030
1027 POKE V + 24,28 : POKE 54273,4 : POKE 54272,73
      : POKE 54276,0 : POKE 54276,129 : GOTO 9
1030 IF A$ < > "N" THEN 1025
1035 POKE V + 22, PEEK ( V + 22 ) AND 239 : END
10000 DATA 255,255,255,255,255,255,255,255,170,170,
      170,170,170,170,170,170
10005 DATA 85,85,85,85,85,85,85,85
10010 DATA 1,1,8,8,16,16,128,128,169,169,162,162,154,
      154,42,42
10015 DATA 86,86,81,81,101,101,21,21,64,64,32,32,4,4,2,2
10020 DATA 106,106,138,138,166,166,168,168,149,149,69,
      69,89,89,84,84
10025 DATA 255,255,255,239,255,255,255,255
20000 DATA 0,0,3,0,0,3,0,0,15,0,0,15,0,0,10,0,128,10,
      2,0,11,20,0,11,22,0,15
20005 DATA 9,128,13,2,160,37,0,154,170,0,38,170,0,9,
      154,0,2,118,8,10,154
20010 DATA 32,170,168,42,170,154,42,128,118,40,0,16,
      0,0,0,99
20015 DATA 0,0,0,0,0,0,192,0,0,192,0,0,128,0,0,128,
      8,0,128,2,0,128,0,80
20020 DATA 192,2,80,192,9,128,112,42,0,170,152,0,170,
      96,0,153,128,0,118,0,0
20025 DATA 154,128,128,170,168,32,154,170,160,118,10,
      160,16,0,160,0,0,0,99
20030 DATA 0,0,0,0,0,0,0,0,0,0,0,0,8,0,128,32,0,32,
      128,0,8,128,0,8,128,32,8
20035 DATA 128,168,8,170,170,168,174,170,232,138,
      154,136,128,168,8,128,32,8
20040 DATA 128,0,8,32,0,32,8,0,128,0,0,0,0,0,0,0,0,99
20045 DATA 106,170,0,154,170,128,166,166,128,154,154,
      160,154,106,168,106
20050 DATA 90,168,170,154,154,166,154,102,169,169,
      100,170,169,153,106,166,106
20055 DATA 170,102,169,166,154,169,105,170,170,
      154,165,170,154,166,170,106,170
20060 DATA 168,170,170,168,170,106,152,153,154,96,0,
      0,0,99
20065 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,40,0,2,170,128,
      10,170,240,46,186,152,190
20070 DATA 250,250,186,250,186,251,250,170,250,254,
      170,174,254,186,170,190,174
20075 DATA 170,186,174,42,186,168,10,250,160,2,250,
```

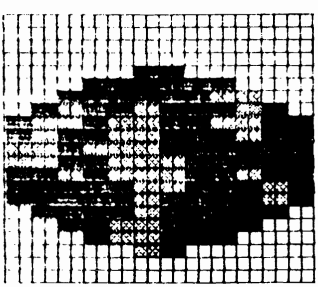
```
         128,0,56,0,0,0,0,0,0,0,99
20080 DATA 0,24,0,0,24,0,0,24,0,0,24,0,0,24,0,0,24,
         0,0,24,0,0,24,0,0,24,0,0,24,0
20085 DATA 0,24,0,0,24,0,0,24,0,0,24,0,0,24,0,0,24,
         0,0,24,0,0,24,0,0,24,0,0,24,0
20090 DATA 0,24,0,99
20095 DATA 0,0,0,0,8,0,0,148,128,0,65,0,1,42,64,2,0,32,
         0,34,0,7,73,112
20100 DATA 112,28,7,7,73,112,0,34,0,2,0,32,1,42,64,
         0,65,0,0,148,128,0,8,0
20105 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,99
```

CHEXSUM

```
   0=4552      125=2687      257=2372
   1=2756      130=677       260=5079
   2=4179      140=3796      265=5346
   3=2827      150=1062      270=3641
   5=4092      155=3322      290=2997
   6=3902      157=1194      291=3371
   8=3596      160=4176      295=4134
   9=3596      162=1457      297=2331
  10=4301      165=5184      300=1741
  15=3784      167=1677      301=3959
  20=3793      170=1208      305=3969
  25=2584      172=3912      310=2628
  30=2575      175=1616      311=3376
  35=3656      180=1110      312=1575
  40=3269      185=4768      315=4409
  45=3270      190=4700      320=4889
  50=3266      195=4087      325=2259
  55=3793      200=4120      350=526
  60=3792      205=4154      500=1765
  65=3791      210=2378      505=2444
  70=3397      215=1494     1000=4332
  75=3385      217=1169     1002=619
  80=3392      218=3924     1003=3376
  85=3393      220=4209     1005=4198
  90=3388      222=2447     1007=1336
  95=3384      225=5087     1010=1294
 100=2217      230=5260     1015=2374
 105=2106      235=945      1020=1395
 110=1356      240=1569     1022=3661
 115=3663      250=1024     1025=1534
 120=4600      251=3959     1027=3911
 121=2025      255=4863     1030=1184
```

```
 1035=1784
10000=3734
10005=1438
10010=2961
10015=2606
10020=3353
10025=1890
20000=3769
20005=3582
20010=2824
20015=3459
20020=3830
20025=3726
20030=3787
20035=3871
20040=2527
20045=3562
20050=4091
20055=4201
20060=2740
20065=4014
20070=4196
20075=3978
20080=4357
20085=4357
20090=596
20095=3363
20100=3577
20105=1896

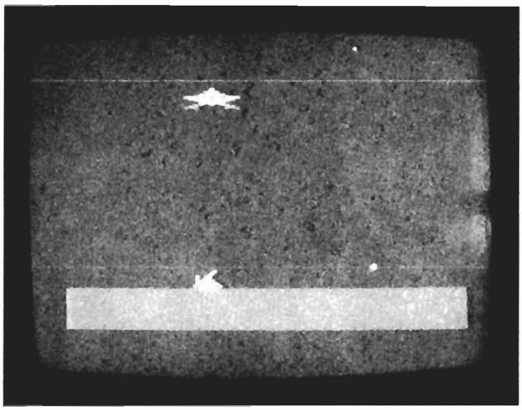
TOTAL= 387009
```



SPRITE

HOME
PLANET

Tank Attack



The aliens are coming again!

Controlling a laser tank, you must try and stop as many starships as possible from landing. If one does land it will release the aliens and fly away.
The starships are protected by a force-field which flickers on and off. When blue the craft can be destroyed, when red the force-field protects the ship. The starships will attempt to destroy you using guided lasers.

Your controls are:
- 'Z' to move the tank left
- 'C' to move the tank right
- 'M' to fire your laser

PROGRAM STRUCTURE	LINES
INITIALIZATION (PRINT SURFACE & SPACE SHIP)	1 — 10
GET KEY PRESSED	15
MOVE LEFT?	20
MOVE RIGHT?	30
FIRE CANNON?	50
HIT SPACE SHIP?	67
MOVE SPACE SHIP	100 — 106
TANK BEING ATTACKED	145 — 170
SPACE SHIP HIT AND FALLING	500 — 520
TANK UNABLE TO PREVENT LANDING	1000 — 1030
TANK HIT BY MISSILE	5000 — 5070
CONTINUE GAME	1040 — 1070

VARIABLES
P = TANK POSITION
A = ALIEN POSITION
PE = KEY PRESSED
B = BOMB POSITION

TANK ATTACK

```
  0 POKE V + 55,255 : POKE V + 56,47 : PRINT "♥" :
      POKE 53281,0 : POKE 53280,0
  1 V = 53248 : POKE 2040,192 : POKE 2041,194 :
      POKE 2042,196 : POKE 2043,197 : POKE 2044,198
  2 POKE V + 39,3 : POKE V + 40,2 : POKE V + 41,2 :
      POKE V + 42,3 : POKE V + 43,12 :
      POKE V + 28,15
  3 POKE V + 37,8 : POKE V + 38,5
  4 FOR I = 0 TO 446 : READ Q : POKE 12288 + I,Q :
      NEXT
  5 FOR I = 56136 TO 56295 : POKE I,9 :
      POKE I - 54272,160 : NEXT : POKE V + 21,31 :
      POKE 54273,0
  6 POKE 54272,0 : POKE 54276,0 : POKE 54277,0 :
      POKE 54278,0 : POKE 54296,15 : POKE 54278,240
  7 POKE 54276,129 : POKE V + 23,0 : POKE V + 29,4
      : P = 110 : POKE V + 1,200 : POKE V + 2,11
  8 POKE V + 5,50 : POKE V + 4,230 :
      A = 150 + INT ( RND ( 1 ) * 70 ) : B = 50 :
      PE = PEEK ( V + 30 ) : G = .4
  9 POKE V + 8,0 : POKE V + 9,220
 10 POKE 54273,0 : PE = PEEK ( V + 30 )
 15 PE = PEEK ( 197 )
 20 IF PE = 12 AND P > 30 THEN P = P - 3 :
      POKE 2040,192 : POKE 54273,2
 30 IF PE = 20 AND P < 320 THEN P = P + 3 :
      POKE 2040,193 : POKE 54273,2
 35 IF P > 255 THEN POKE V,P - 256 : POKE V + 16,
      PEEK ( V + 16 ) OR 1 : GOTO 50
 40 POKE V,P : POKE V + 16, PEEK ( V + 16 ) AND 254
 50 IF PE < > 36 THEN 100
 60 FOR I = 50 TO 0 STEP - 2 : POKE 54273,I : NEXT
 62 X = P - 5 : Y = 192 : D = - 1 : POKE 2041,195 :
      IF PEEK ( 2040 ) = 193 THEN D = 1 :
      X = X + 20 : POKE 2041,194
 65 IF PEEK ( V + 30 ) < > 6 THEN 70
 66 POKE V + 2,0 : POKE V + 3,190 : X = 30 :
      FOR I = 0 TO 10 : POKE 54273,I : NEXT
 67 IF PEEK ( V + 41 ) = 246 THEN GOSUB 500
 68 POKE 54273,0
 70 IF X < 40 OR X > 300 OR Y < 50 THEN X = 0 :
      Y = 0 : POKE V + 2,0 : POKE V + 3,192 :
      POKE 54273,0 : GOTO 100
 80 X = X + D * 5 : Y = Y - 6
 85 POKE V + 3,Y : IF X > 255 THEN
      POKE V + 2,X - 256 : POKE V + 16,
      PEEK ( V + 16 ) OR 2 : GOTO 95
 90 POKE V + 2,X : POKE V + 16, PEEK ( V + 16 )
      AND 253
 95 GOTO 65
100 G = G + .02 : IF RND ( 1 ) < .6 THEN A = A - 3
101 B = B + G : IF RND ( 1 ) < .1 THEN A = A + 4
102 IF A > 320 THEN A = 240
105 IF A < 40 THEN A = 40
```

```
106 IF A > 255 THEN POKE V + 4,A - 256 :
      POKE V + 16, PEEK ( V + 16 ) OR 4 : GOTO 110
107 POKE V + 4,A : POKE V + 16, PEEK ( V + 16 )
      AND 251
110 POKE V + 5,B : IF RND ( 1 ) < .3 THEN
      POKE V + 41,2
115 IF RND ( 1 ) < .1 THEN POKE V + 41,6
120 IF B > 185 THEN 1000
130 IF RND ( 1 ) < .96 THEN GOTO 10
140 IF A < P THEN POKE 2041,195
142 IF A > P THEN POKE 2041,194
145 FOR I = 0 TO 40 STEP .4 : POKE 54273,I : NEXT :
      POKE 54273,0
146 SH = A : FOR I = B TO 215 STEP 6
150 IF RND ( 1 ) < .4 THEN
      SH = SH + ( ( SH > P ) - ( SH < P ) ) * 5
160 IF PEEK ( V + 30 ) = 3 THEN 5000
161 IF SH > 255 THEN POKE V + 2,SH - 256 :
      POKE V + 16, PEEK ( V + 16 ) OR 2 : GOTO 165
162 POKE V + 2,SH : POKE V + 16, PEEK ( V + 16 )
      AND 253
165 POKE V + 3,I
170 NEXT
180 POKE V + 2,0 : POKE V + 3,190
190 GOTO 10
200 A = INT ( RND ( 1 ) * 50 ) + 170 : B = 50 :
      X = 30 : POKE V + 29,4 : POKE V + 23,4
205 POKE V + 16, PEEK ( V + 16 ) AND 251
500 POKE V + 2,0 : POKE V + 3,192
502 POKE 54273, ( B - 40 )
505 G = G + .02 : B = B + G : POKE V + 5,B :
      IF B < 205 THEN 502
506 POKE 54273,2 : POKE V + 29,20 :
      POKE V + 9,B - 5 : POKE V + 4,0 :
      POKE V + 5,0
507 IF A > 255 THEN POKE V + 8,A - 256 :
      POKE V + 16, PEEK ( V + 16 ) OR 16 :
      GOTO 510
508 POKE V + 8,A : POKE V + 16, PEEK ( V + 16 )
      AND 239
510 FOR Z = 10 TO 0 STEP - .2 : POKE 54273,I :
      NEXT : A = 150 + INT ( RND ( 1 ) * 70 ) :
      B = 50 : POKE 54273,0
515 FOR Z = 1 TO 500 : NEXT : POKE V + 29,4 :
      POKE V + 8,0 : POKE V + 9,220
517 IF A > 255 THEN POKE V + 4,A - 256 :
      POKE V + 16, PEEK ( V + 16 ) OR 4 : GOTO 520
518 POKE V + 4,A : POKE V + 16, PEEK ( V + 16 )
      AND 251
520 POKE V + 5,B : G = 0 : RETURN
1000 POKE V,0 : POKE V + 1,0 : POKE V + 2,0 :
      POKE V + 3,0 : POKE 54276,0 : POKE 54276,17
1001 POKE V + 8,0 : POKE V + 9,0
1005 PRINT "{S}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}
      {Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{Q}{↑}{R}{]}{]}{]}{]}{]}{]}{]}{]}{]}
      THE ▲ ALIENS ▲ HAVE ▲ INVADED"
```

```
1006 A = A + 10 : FOR T = 1 TO 3 : IF A > 255 THEN
       POKE V + 6,A - 256 : POKE V + 16,
       PEEK ( V + 16 ) OR 8 : GOTO 1010
1007 POKE V + 6,A : POKE V + 16, PEEK ( V + 16 )
       AND 247
1010 FOR I = B TO 205 : POKE V + 7,I :
       POKE 54273,I - 185 : NEXT I
1011 FOR I = A TO 0 STEP - 1
1012 POKE V + 6,I : POKE V + 16, PEEK ( V + 16 )
       AND 247
1015 POKE 54273,I : NEXT I,T
1020 FOR I = B TO 20 STEP - 1 : POKE V + 5,I :
       POKE 54273, INT ( RND ( 1 ) * 200 ) : NEXT
1025 POKE 54273,0 : POKE 54276,0 : POKE 54276,33
1030 FOR I = 50 TO 1 STEP - .4 : POKE 54273,I : NEXT
       : POKE 54273,0
1040 PRINT "{CLR}{DOWN}{DOWN}{DOWN}{DOWN}PLAY AGAIN?"
1050 GET A$ : IF A$ = "Y" THEN RUN
1060 IF A$ < > "N" THEN 1050
1070 END
5000 POKE 54276,0 : POKE 54276,129 : POKE V + 2,0 :
       POKE V + 3,190
5001 POKE V,0 : POKE V + 1,0 : POKE V + 9,200
5002 IF P > 255 THEN POKE V + 8,P - 256 :
       POKE V + 16, PEEK ( V + 16 ) OR 16 : GOTO 5004
5003 POKE V + 8,P : POKE V + 16, PEEK ( V + 16 )
       AND 239
5004 FOR I = 20 TO 0 STEP -.1 : POKE 54273,I : NEXT
       : POKE 54273,0
5005 POKE 54276,0 : POKE 54276,33
5010 FOR I = 200 TO 0 STEP - 2 : POKE 54273,I : NEXT
5020 FOR I = 0 TO 10 STEP .2 : POKE 54273,I : NEXT
5025 POKE 54273,0 : POKE 54276,0 : POKE 54276,17
5030 FOR I = 10 TO 30 STEP .1 : POKE 54273,I : NEXT
5040 FOR I = 50 TO 0 STEP - .4 : POKE 54273,I : NEXT
5050 POKE 54273,0
5060 B = B + 1 : POKE V + 5,B : POKE 54273,B :
       IF B < 185 THEN 5060
5065 POKE V,0 : POKE V + 1,0 : POKE 54273,0
5070 GOTO 1000
9000 DATA 0,0,0,0,0,0,0,0,0,2,0,0,0,128,0,0
       ,128,0,0,32,0,0,8,8,0,9,8,0,21,8,0
9010 DATA 21,8,0,85,72,0,85,72,3,255,248,15
       ,255,252,51,51,51,51,51,51,15,255,252
9020 DATA 3,255,240,0,0,0,0,0,0,99
9030 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,32,0
       ,0,128,0,0,128,0,2,0,8,9,0,8,21,0,8
9040 DATA 21,0,8,85,64,8,85,64,11,255,240
       ,15,255,252,51,51,51,51,51,51,15,255
9050 DATA 252,3,255,240,0,0,0,0,0,0,99
9060 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
9070 DATA 0,10,0,0,38,0,0,38,0,0,38,0,0
       ,40,0,0,32,0,0,128,0
9080 DATA 0,0,0,0,0,0,0,0,0,0,0
       ,0,0,0,0,0,0,0,0,0,0,99
9090 DATA 0,0,0,0,0,0,0,0,0,0,0
```

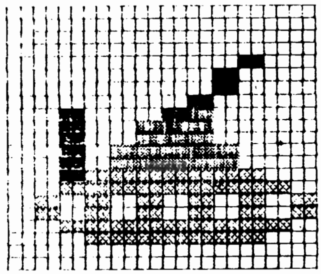
```
     ,0,0,0,0,0,0,0,0,0,0
9100 DATA 0,160,0,0,152,0,0,152,0,0,152
     ,0,0,40,0,0,8,0,0,2,0
9110 DATA 0,0,0,0,0,0,0,0,0,0,0,0
     ,0,0,0,0,0,0,0,0,0,99
9120 DATA 0,48,0,0,48,0,0,252,0,0,48,0,3,87
     ,0,9,101,128,252,100,124,61,101,240
9130 DATA 15,155,192,3,255,0,1,253,0,1,49
     ,0,4,0,64,16,0,16,136,0,136,0,0,0,0,0,0
9140 DATA 0,0,0,0,0,0,0,0,0,0,0,0,99
9150 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
9160 DATA 0,4,0,0,29,0,0,29,0,0
     ,4,0,0,4,0,0,34,0,0,34,0
9170 DATA 0,0,0,0,0,0,0,0,0,0,0,0
     ,0,0,0,0,0,0,0,0,0,99
9180 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
9181 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
9182 DATA 0,0,0,0
9190 DATA 3,7,0,7,223,184,15,255,248,15
     ,255,252,63,255,255,255,255,255
9200 DATA 255,255.255,255,255,255,255,255,255
```

CHEXSUM

0=3255	85=4668	200=3856
1=3935	90=2306	205=1572
2=4408	95=477	500=1338
3=1342	100=2251	502=926
4=2221	101=2150	505=3048
5=4292	102=1157	506=3592
6=3836	105=1033	507=3976
7=4155	106=3921	508=2287
8=5152	107=2279	510=4771
9=1345	110=2240	515=3147
10=1640	115=1582	517=3920
15=796	120=951	518=2279
20=3202	130=1490	520=1164
30=3254	140=1250	1000=3590
35=3689	142=1249	1001=1218
40=2040	145=2505	1005=3754
50=1115	146=1536	1006=5560
60=1984	150=3327	1007=2287
62=5445	160=1456	1010=2780
65=1525	161=4139	1011=1074
66=3503	162=2393	1012=2295
67=1635	165=587	1015=954
68=528	170=131	1020=3595
70=5454	180=1338	1025=1832
80=1492	190=472	1030=2686

```
1040=1222
1050=1263
1060=1183
1070=129
5000=2711
5001=1730
5002=4068
5003=2302
5004=2683
5005=1219
5010=2038
5020=1860
5025=1836
5030=1912
5040=2031
5050=528
5060=2943
5065=1587
5070=579
9000=4158
9010=4562
9020=1442
9030=4138
9040=4183
9050=1710
9060=2329
9070=2792
9080=2498
9090=2329
9100=2844
9110=2498
9120=4270
9130=4485
9140=1533
9150=2329
9160=2581
9170=2498
9180=1830
9181=1830
9182=482
9190=3497
9200=2089
```

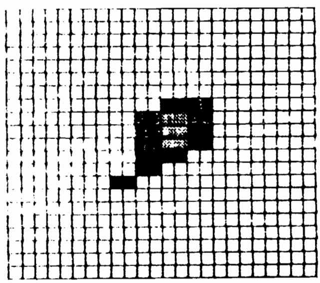
TOTAL= 282363



SPRITE

PLAYER'S TANK

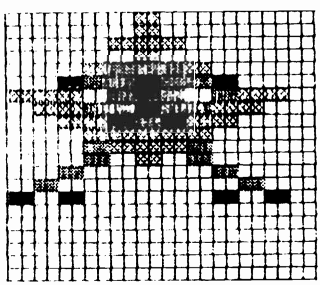
SPRITE # 1



SPRITE

PLAYER'S MISSILE

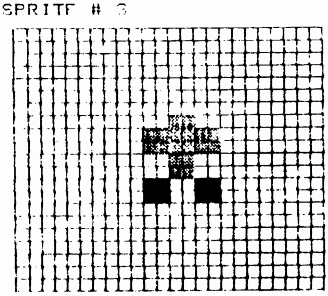
SPRITE # 2



SPRITE

ALIEN'S SHIP

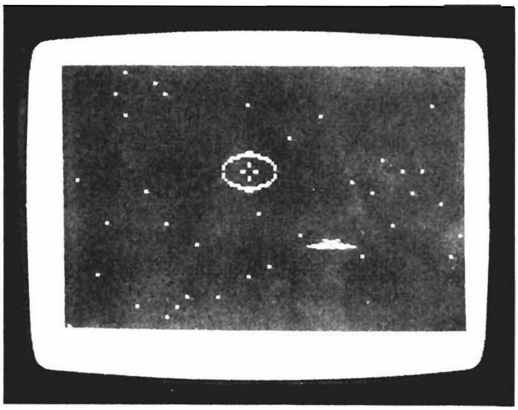
SPRITE # 3



SPRITE

ALIEN

Star Pilot



This program also contains some machine code routines.

In this game you, the Star Pilot, must attempt to get approaching and retreating enemy space craft into your sights and blast them.

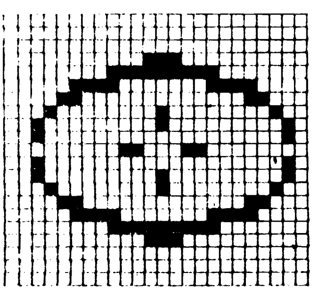
The game's objective is to get as many enemy craft as possible within the 2 minutes playing time allowed.

'W' and 'X' control your up and down movement.
'A' and'D' control your left and right travel.
'S' key is used to fire with.

PROGRAM STRUCTURE	LINES
SET UP COLOUR MODE: TURN SOUND ON	1
READ IN SPRITE DATA	5
SPRITE 0 = SIGHTS SPRITE 1 = LEFT MISSILE SPRITE 2 = RIGHT MISSILE SPRITE 3 = ALIEN PUT SIGHTS ON SCREEN	10
GIVE X, Y COORDINATES OF SIGHTS (148, 108) MAKE MISSILES MULTI-COLOUR MAKE MISSILES RED	15
CLEAR SCREEN: GIVE ADDRESS OF KEYBOARD	20
LOAD IN MACHINE LANGUAGE ROUTINE	25
GET ADDRESSES OF THE 3 MACHINE CODE ROUTINES SET CHARACTER COLOUR TO WHITE	30
PLACE STARS RANDOMLY IN BACKGROUND	35
INITIALIZE TIME = 0 SCORE = 0	40
READ KEYBOARD GET SPRITE COLLISION REGISTER MAKE SIGHTS GREEN IF SIGHTS AND ALIEN ARE TOUCHING THEN MAKE SIGHTS RED	50
IF PLAYER DID NOT HIT 'FIRE' THEN 100	52
SET UP VOICE 1 ON SID CHIP	53
PUT L AND R MISSILES ON SCREEN	55 — 60
MOVE MISSILES TOWARDS SIGHTS	65 — 75

IF BOTH L AND R MISSILE DID NOT HIT ANYTHING THEN 85	75 — 80
IF ALIEN WAS HIT THE 1000	82
MAKE SMALL EXPLOSION SOUND	83
PUT EXPLOSION SPRITE INSTEAD OF MISSILE ON SCREEN, WAIT, THEN ERASE	85 — 90
RESTORE POINTER TO MISSILE AGAIN	95
IF ALIEN IS ON SCREEN THEN 130	100
90% CHANCE EACH TIME ROUND LOOP, OF NOT CREATING AND ALIEN IF ONE ALREADY THERE. NO ALIEN TO BE CREATED THEN 200	105
SET RANDOM X, Y, Z OF ALIEN FROM SPACE-SHIP. THEN PUT ALIEN ON SCREEN, SIZE OF ALIEN DEPENDENT ON Z. AREAS 195 — 199 OF MEMORY CONTAIN DEFINITIONS FOR ALIEN SPRITE. 195 — SMALLEST. 199 — LARGEST. SET ALIEN PRESENT STATUS FLAG.	107 — 129
MOVE ALIEN IN X, Y, Z DIRECTION RANDOMLY	130 — 175
IF PLAYER DID NOT HIT DIRECTION KEY THEN 300. IF NO ALIEN PRESENT THEN 232	200
MOVE ALIEN RELATIVE TO OUR SPACE SHIP IF WE ARE GOING UP THEN MOVE ALIEN DOWN, IF WE ARE GOING LEFT THEN MOVE ALIEN RIGHT AND VICE VERSA	205 — 230
IF UP DIRECTION SELECTED, THEN MAKE STAR FIELD MOVE DOWN RELATIVE TO SPACE SHIP. PLACE STAR RANDOMLY AT TOP OF SCREEN. GO TO 300.	232

IF LEFT DIRECTION THEN MOVE STAR FIELD RIGHT. PLACE STAR RANDOMLY ALONG RIGHT-MOST COLUMN. GO TO 300.	235
IF RIGHT DIRECTION THEN MOVE STAR FIELD LEFT. PLACE STAR RANDOMLY ALONG LEFT-MOST COLUMN. GO TO 300.	240
SCROLL SCREEN UP FOR DEFAULT DOWN DIRECTION. PLACE A STAR RANDOMLY ALONG BOTTOM LINE	245 — 250
IF TIME IS NOT UP THEN 50 ASK FOR NEW GAME	300 — 305
IF REPLY = YES THEN GO TO 30	320
OTHERWISE END	325
BLOW UP ALIEN. MAKE BIG EXPLOSION SOUNDS, PLACE EXPLOSION CLOUDS ON SCREEN. SET ALIEN PRESENT FLAG = 0	1000 — 1025
SPRITE DATA	10000 — 10130
DOWN DIRECTION SCROLLING ROUTINE	20000 — 20010
RIGHT SCROLLING ROUTINE	30000 — 30010
LEFT SCROLLING ROUTINE	40000 — 40010



SPRITE

SIGHTS

STAR PILOT

```
  1 POKE 55,255 : POKE 56,47 : V = 53248 :
      POKE V + 32,12 : POKE V + 33,0 :
      POKE V + 38,5 : POKE 54296,15
  5 FOR I = 0 TO 638 : READ A : POKE 1024 * 12 + I,A
      : NEXT
 10 POKE 2040,192 : POKE 2041,193 : POKE 2042,194 :
      POKE V + 21,1 : POKE V + 39,5 : POKE V + 23,1
 15 POKE V + 29,9 : POKE V,148 : POKE V + 1,108 :
      POKE V + 28,6 : POKE V + 40,2 : POKE V + 41,2
      : POKE V + 37,8
 20 KB = 197 : PRINT "♥"
 25 FOR I = 0 TO 151 : READ A : POKE 13 * 1024 + I,A
      : NEXT
 30 D = 13 * 1024 : LE = D + 52 : RI = D + 102 :
      FOR I = 55296 TO 56295 : POKE I,1 : NEXT
 35 FOR I = 1 TO 50 :
      POKE 1024 + RND ( 1 ) * 1000,46 : NEXT
 40 AL = 0 : TI$ = "000000" : SC = 0
 50 K = PEEK ( KB ) : A = PEEK ( V + 30 ) :
      POKE V + 39,5 : IF A = 9 THEN POKE V + 39,2
 52 IF K < > 13 THEN 100
 53 POKE 54276,0 : POKE 54273,72 : POKE 54272,169 :
      POKE 54277,143 : POKE 54276,129
 55 POKE V + 2,60 : POKE V + 3,220 : POKE V + 5,220
 60 POKE V + 4,4 : POKE V + 16, PEEK ( V + 16 ) OR 4
      : POKE V + 21, PEEK ( V + 21 ) OR 6
 65 FOR I = 1 TO 32 : POKE V + 2, PEEK ( V + 2 ) + 3
      : POKE V + 3, PEEK ( V + 3 ) - 3 :
      POKE V + 5, PEEK ( V + 3 )
 70 A = PEEK ( V + 4 ) - 3 : IF A < 0 THEN A = 255 :
      POKE V + 16, PEEK ( V + 16 ) - 4
 72 POKE V + 4,A
 75 A = PEEK ( V + 30 ) : NEXT : POKE V + 21,
      PEEK ( V + 21 ) - 6 : B = A AND 2 :
      IF B = 0 THEN B = A AND 4
 80 IF B = 0 THEN 85
 82 IF A AND 8 THEN 1000
 83 POKE 54276,0 : POKE 54273,17 : POKE 54277,7 :
      POKE 54276,129
 85 POKE V + 2, PEEK ( V ) + 10 : POKE V + 3,
      PEEK ( V + 1 ) + 10 : POKE 2041,201 :
      POKE V + 21, PEEK ( V + 21 ) OR 2
 90 FOR I = 1 TO 50 : NEXT : POKE V + 21,
      PEEK ( V + 21 ) - 2
 95 POKE 2041,193
100 IF AL > 0 THEN 130
105 IF RND ( 1 ) > .1 THEN 200
107 IF PEEK ( V + 16 ) AND A THEN POKE V + 16,
      PEEK ( V + 16 ) - 8
110 POKE V + 6, RND ( 1 ) * 256 : IF RND ( 1 ) > .75
      THEN POKE V + 6, RND ( 1 ) * 63 : POKE V + 16,
      PEEK ( V + 16 ) OR 8
115 POKE V + 7, RND ( 1 ) * 200 : Z = INT
      ( RND ( 1 ) * 5 ) : POKE V + 42,4
```

```
120 DZ = INT ( RND ( 1 ) * 5 - 2 ) : DY = INT
    ( RND ( 1 ) * 5 - 2 ) : DX = INT
    ( RND ( 1 ) * 5 - 2 )
121 IF ( DX = 0 AND DY = 0 AND DZ = 0 ) OR
    ( DX = 0 AND DY = 0 AND DZ > 0 ) THEN 120
122 IF Z = 0 THEN 129
123 IF PEEK ( V + 16 ) AND 8 THEN POKE V + 16,
    PEEK ( V + 16 ) - 8
124 IF RND ( 1 ) > .5 THEN 127
125 POKE V + 6,2 : IF RND ( 1 ) > .5 THEN
    POKE V + 6,86 : POKE V + 16, PEEK ( V + 16 )
    OR 8
126 GOTO 129
127 POKE V + 7,31 : IF RND ( 1 ) > .5 THEN
    POKE V + 7,246
129 POKE 2043,195 + Z : POKE V + 21, PEEK ( V + 21 )
    OR 8 : AL = 1
130 A = PEEK ( V + 7 ) + DY * 2 :
    IF A < 31 OR A > 246 THEN AL = 0 :
    POKE V + 21, PEEK ( V + 21 ) - 8 : GOTO 200
135 POKE V + 7,A : A = PEEK ( V + 6 ) + DX * 2 :
    B = PEEK ( V + 16 ) AND 8
140 IF ( A < 2 AND B = 0 ) OR ( A > 86 AND B ) THEN
    AL = 0 : POKE V + 21, PEEK ( V + 21 ) - 8 :
    GOTO 200
145 IF A < 0 THEN A = 255 : POKE V + 16,
    PEEK ( V + 16 ) - 8
150 IF A > 255 THEN A = 0 : POKE V + 16,
    PEEK ( V + 16 ) OR 8
155 POKE V + 6,A : Z = Z + DZ / 20 : IF Z < 0 THEN
    AL = 0 : POKE V + 21, PEEK ( V + 21 ) - 8 :
    GOTO 200
157 IF Z > 4.9 THEN Z = 4.9 : DZ = - INT
    ( RND ( 1 ) * 3 )
160 POKE 2043,195 + Z
165 IF RND ( 1 ) < .9 THEN 200
170 DX = INT ( RND ( 1 ) * 5 - 2 ) :
    DY = INT ( RND ( 1 ) * 5 - 2 ) :
    IF RND ( 1 ) > .7 THEN DZ = INT
    ( RND ( 1 ) * 5 - 2 )
175 IF DX = 0 AND DY = 0 AND DZ = 0 THEN 170
200 IF K < > 9 AND K < > 18 AND K < > 23
    AND K < > 10 THEN 300
203 IF AL = 0 THEN 232
205 A = PEEK ( V + 7 ) - ( K = 9 ) * 8 +
    ( K = 23 ) * 8
206 IF A < 31 OR A > 246 THEN AL = 0 : POKE V + 21,
    PEEK ( V + 21 ) - 8 : GOTO 50
207 POKE V + 7,A
210 B = PEEK ( V + 16 ) AND 8 : A = PEEK ( V + 6 )
    - ( K = 10 ) * 8 + ( K = 18 ) * 8
215 IF ( A < 2 AND B = 0 ) OR ( A > 86 AND B )
    THEN AL = 0 : POKE V + 21, PEEK ( V + 21 ) - 8
    : GOTO 200
220 IF A < 0 THEN A = 255 : POKE V + 16,
    PEEK ( V + 16 ) - 8
```

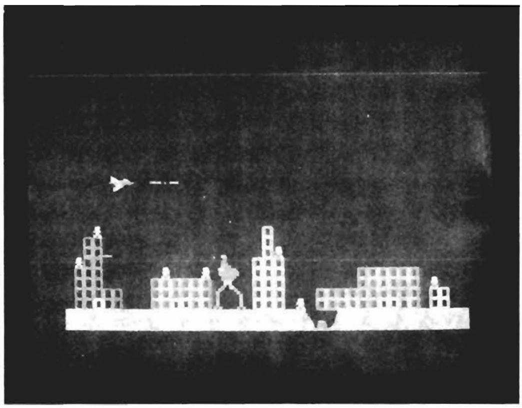
```
225 IF A > 255 THEN A = 0 : POKE V + 16,
      PEEK ( V + 16 ) OR 8
230 POKE V + 6,A
232 IF K = 9 THEN SYS D : POKE 1024 +
      RND ( 1 ) * 40,46 : GOTO 300
235 IF K = 10 THEN SYS RI : POKE 1024 + INT
      ( RND ( 1 ) * 25 ) * 40,46 : GOTO 300
240 IF K = 18 THEN SYS LE : POKE 1063 +
      INT ( RND ( 1 ) * 25 ) * 40,46 : GOTO 300
245 PRINT "{CLR}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}
      {DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{E}▲▲▲▲▲
      ▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲
      ▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲";
250 POKE 56295,1 : POKE 1984 + RND ( 1 ) * 40,46
300 IF TI$ < "000200" THEN 50
305 PRINT "{CLR}◣TIME▲UP{DOWN}{DOWN}{DOWN}{DOWN}" : PRINT "{SYM}YOU
      ▲SCORED";SC : POKE V + 27,255
310 PRINT "{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}WOULD▲YOU▲LIKE▲ANOTHER
      ▲GAME?"
315 GET A$ : IF A$ = "Y" THEN POKE V + 21,1 :
      PRINT "{SYM}" : POKE V + 27,0 : GOTO 30
320 IF A$ < > "N" THEN 315
325 END
1000 POKE V + 21,9 : AL = 0 : POKE V + 2,
      PEEK ( V ) + 10 : POKE V + 3,
      PEEK ( V + 1 ) + 10
1001 POKE 54276,0 : POKE 54273,8 : POKE 54272,47 :
      POKE 54277,76 : POKE 54276,129
1005 POKE V + 4, PEEK ( V ) : POKE V + 5,
      PEEK ( V + 1 ) : POKE V + 23, PEEK ( V + 23 )
      OR 4
1010 POKE V + 29, PEEK ( V + 29 ) OR 4 :
      POKE 2041,200 : POKE 2042,200
1015 POKE V + 21, PEEK ( V + 21 ) OR 6 :
      FOR A = 200 TO 201 : POKE 2042,A :
      FOR B = 200 TO 201.9 STEP .05
1020 POKE 54276,0 : POKE 54276,129 : POKE 2041,B :
      NEXT B,A : POKE V + 23,1 : POKE V + 29,9
1025 POKE V + 21,1 : POKE 2041,193 : POKE 2042,194 :
      SC = SC + ( 5 - INT ( Z ) ) * 10 : GOTO 100
10000 DATA 0,0,0,0,0,0,0,0,0,0,28,0,0,255,128,7
      ,128,240,12,0,24,16,8,4,32,8,2
10005 DATA 32,0,2,0,99,0,32,0,2,32,8,2,16,8,4
      ,12,0,24,7,128,240,0,255,128,0,28,0
10010 DATA 0,0,0,0,0,0,0,0,0,99
10015 DATA 0,0,0,0,0,0,0,2,128,0,2,128,0,41,128
      ,0,41,128,0,150,0,0,150,0,2,86,0
10020 DATA 2,86,0,2,88,0,2,88,0,9,160,0,9,160,0
      ,38,0,0,38,0,0,40,0,0,40,0,0
10025 DATA 128,0,0,128,0,0,0,0,0,99
10030 DATA 0,0,0,0,0,0,2,128,0,2,128,0,2,104
      ,0,2,104,0,0,150,0,0,150,0
10035 DATA 0,149,128,0,149,128,0,37,128,0,37,128
      ,0,10,96,0,10,96,0,0,152,0,0,152
10040 DATA 0,0,40,0,0,40,0,0,2,0,0,2,0,0,0,99
```

```
10045 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
      ,0,0,0,0,0,0,0,0,8,0,0,62,0,0,0,0
10050 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
      ,0,0,0,0,0,0,0,0,0,0,0,0,99
10055 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
      ,0,0,0,0,0,8,0,0,28,0,0,255,128
10060 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
      ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,99
10065 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
      ,0,0,34,0,0,28,0,0,127,0,3,255,224
10070 DATA 0,201,128,0,0,0,0,0,0,0,0,0,0,0,0
      ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
10075 DATA 99,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
      ,99,0,0,28,0,0,62,0,0,247,128
10080 DATA 7,255,240,31,255,252,1,201,192,0,0,0
      ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
10085 DATA 0,0,0,0,0,0,0,99
10090 DATA 0,0,0,0,0,0,0,0,0,0,0,0,128,128,0
      ,99,0,0,28,0,0,127,0,1,227,192,15
10095 DATA 255,248,127,255,255,7,156,240,2,0
      ,32,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
10100 DATA 0,0,0,0,0,0,0,99
10105 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,32
      ,0,0,186,128,10,247,128,47,123,128
10110 DATA 181,151,224,173,85,120,11,95,248,2
      ,222,160,2,222,0,2,222,0,0,190,0
10115 DATA 0,40,0,0,0,0,0,0,0,0,0,0,0,0,99
10120 DATA 0,0,0,0,0,0,0,2,128,0,138,232,2,239
      ,120,11,117,224,45,119,168,47,87
10125 DATA 254,10,213,94,11,85,248,11,87,160
      ,45,221,128,181,221,224,183,223,120
10130 DATA 46,187,94,8,34,222,0,0,184,0,0
      ,32,0,0,0,0,0,0,0,0,0
20000 DATA 162,192,189,255,6,157,39,7,202,208
      ,247,202,189,255,5,157,39,6,202,208
20005 DATA 247,202,189,255,4,157,39,5,202,208
      ,247,202,189,255,3,157,39,4,202,208
20010 DATA 247,169,32,157,0,4,232,224,40,208,248,96
30000 DATA 169,0,133,20,169,4,133,21,160,1,177
      ,20,136,145,20,200,200,192,40
30005 DATA 208,245,136,169,32,145,20,169,40,24
      ,101,20,133,20,169,0,101,21,133,21
30010 DATA 201,7,208,221,165,20,201,232,208,215,96
40000 DATA 169,0,133,20,169,4,133,21,160,39,177
      ,20,200,145,20,136,136,192,255
40005 DATA 208,245,200,169,32,145,20,169,40,24
      ,101,20,133,20,169,0,101,21,133,21
40010 DATA 201,7,208,221,165,20,201,232,208,215,96
```

CHEXSUM

1=4616	145=2663	10025=1438
5=2438	150=2653	10030=3358
10=4187	155=5029	10035=4424
15=4981	157=2703	10040=2003
20=946	160=839	10045=3980
25=2432	165=1106	10050=3527
30=4314	170=5554	10055=3764
35=2504	175=2052	10060=4222
40=1744	200=3239	10065=4127
50=3848	203=860	10070=4341
52=1021	205=2592	10075=3712
53=3441	206=3900	10080=4470
55=2151	207=579	10085=960
60=3779	210=4004	10090=4266
65=5010	215=4854	10095=4496
70=3801	220=2663	10100=960
72=576	225=2653	10105=4256
75=4748	230=578	10110=4073
80=720	232=2967	10115=1950
82=831	235=3723	10120=4165
83=2568	240=3721	10125=4280
85=5103	245=3205	10130=2905
90=2544	250=2006	20000=4468
95=574	300=1235	20005=4461
100=857	305=3480	20010=2374
105=1106	310=2706	30000=3861
107=2777	315=3551	30005=4427
110=5182	320=1122	30010=2297
115=3168	325=129	40000=4089
120=4534	1000=3989	40005=4424
121=4059	1001=3284	40010=2297
122=794	1005=3734	
123=2760	1010=2797	TOTAL= 347791
124=1114	1015=4559	
125=3864	1020=3872	
126=535	1025=4414	
127=2348	10000=4030	
129=2941	10005=4356	
130=5539	10010=1190	
135=3453	10015=4257	
140=4854	10020=3806	

Martian Invasion



Fly your plane over the city, watching Martian Pods drop from the sky.

A tripod or a flying machine will hatch out from the crater left by the Pod. The tripods shoot at you and the flying machine will try to crash into you. All this while a red weed starts growing and crawling.

You must shoot as many Martians as possible before your inevitable defeat. If you crash into a pod you explode, so avoid them. This game has excellent graphics and should provide many hours of fun.

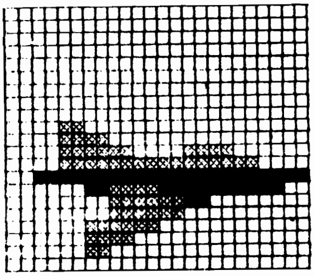
Your controls are:

'W' up
'X' down
'A' left
'D' right
'S' to fire your laser

See how many you can get!

PROGRAM STRUCTURE	LINES
INITIALIZATION	0 — 10
PRINT GROUND/BUILDING	15 — 45
INITIALIZE VARIABLES/COLOURS	50 — 53
GET KEY PRESSED	55
FACE FIGHTER IN LEFT/RIGHT DIRECTION	56 — 58
CHECK HORIZONTAL POSITION OF CRAFT	60 — 75
MOVE CRAFT TO NEW HORIZONTAL/VERTICAL POSITION	80 — 95
PUT POD ON SCREEN	110 — 115
PUT TRIPOD ON SCREEN	150 — 170
PRINT FLYING MACHINE	250 — 270
EXPLODE FLYING MACHINE	300
FIGHTER COLLIDED WITH ANYTHING?	500
SHELL FIRED?	505
MOVE SHELL	510 — 535
COLLISION OCCURRED?	675
EXPLOSION SOUND OF ALIEN	677 — 690
DESTROY FIGHTER	1000 — 1005
DECREMENT NO. OF LIVES LEFT — FINISHED?	1035 — 1045
PLAY AGAIN?	1035 — 1045
END	1060

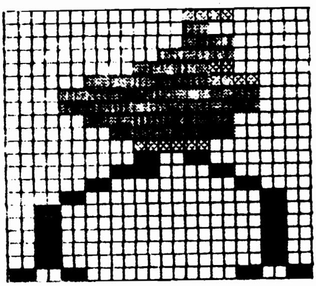
VARIABLES
CO = DIFFERENCE BETWEEN SCREEN AND VIDEO MEMORY
V = START OF VIDEO CHIP
A = HOLDS DATA READ IN
K = CONTAINS LAST KEY PRESSED
B = HIGH ORDER BIT FOR SCREEN ADDRESSING
A = POSITION OF FIGHTER CRAFT
MA = MARTIAN TYPE: 1 = POD; 2 = TRIPOD; 3 = FLYING MACHINE;
0 = LANDED; −1 = NOT EXISTENT
X,Y = POSITIONS FOR SPRITES ON SCREEN
KB = POINTER TO LAST KEY PRESSED REGISTER
SC = SCORE
SH = SHELL FIRED FLAG
I = VARIABLE



SPRITE

PLAYER'S
SHIP
MOVING
RIGHT

SPRITE # 1



sprite

ALIEN
MOVING
RIGHT

MARTIAN INVASION

```
 0 POKE 55,255 : POKE 56,47 : V = 53248 :
     FOR I = 12544 TO 12551 : POKE I,0 : NEXT :
     CO = 54272
 1 POKE 54296,15
 2 POKE V + 34,11 : POKE V + 35,8 : POKE V + 32,0 :
     POKE V + 33,0 : PRINT "♥";
 5 FOR I = 13312 TO 14015 : READ A : POKE I,A :
     NEXT : POKE V + 37,11 : POKE V + 38,5
10 FOR I = 12288 TO 12375 : READ A : POKE I,A :
     NEXT : POKE V + 22, PEEK ( V + 22 ) OR 16 :
     POKE V + 24,28
15 PRINT "◙◙◙◙◙◙◙◙◙◙◙◙◙◙◙◙◙◙
     ◙◙◙◙◙◙◘BBBBBBBBBBBBBBBBBBBBBBBBBBBB
     BBBBBBBBBBB";
20 PRINT "BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB"
     ; : POKE 2023,2 : POKE 56295,5
25 PRINT "□□□◘◘◘əə◘ə◘əə□■■■■■ə
     əəə□■■■■■əəə□■■■əəə□■■■◘
     A◘əə□■■əə□■■əə□■◘A";
30 PRINT "◘◘◘◘◘◘◙◙◙◙A▲ ▲ ▲A◙■■■
     ■■■◘əəəəəə◙■■■■■■əəəəəə◙■■■■
     ■■əə◘əə◘əə";
35 PRINT "◘◘◘◘▓əəə□■■■əəə□■■■əəə□
     ■■■əəə□■■■əəə□■■ə◘A□■■▓ə□
     ■ə◙◙◙◙◙◙◙";
40 PRINT "◘◘◘A▲◘əəəəəə◘əə◘əə□■■■■■
     ■■■■■əəəəəəəəəə□■■■■■■əəəəəə□■
     ■■■■■əəəəəə";
45 PRINT "◙◙◙◘◘əə□■■əə□■■◘A";
50 POKE V + 28,3 : POKE V + 39,2 : POKE V,20 :
     POKE V + 1,99 : KB = 197 : POKE V + 16,0 :
     POKE 2042,215
52 POKE 2040,216 : DI = 0 : POKE V + 21,1 :
     MA = - 1 : PO = PEEK ( V + 30 ) : PO = 0 :
     SH = 0 : RW = 0 : M1 = 0
53 SC = 0 : LI = 3 : PS = 0 : POKE 54290,0 :
     POKE 54291,0 : POKE 54292,240
55 K = PEEK ( 197 ) : B = PEEK ( V + 16 ) AND 1 :
     A = PEEK ( V ) + ( K = 10 ) * 8 -
     ( K = 18 ) * 8
56 IF K = 10 THEN DI = 1
57 IF K = 18 THEN DI = 0
58 POKE 2040,216 + DI
60 IF A < 0 AND B THEN POKE V + 16,
     PEEK ( V + 16 ) AND 254 : B = 0 : A = 248
65 IF A > 255 AND B = 0 THEN POKE V + 16,
     PEEK ( V + 16 ) OR 1 : B = 1 : A = 0
70 IF A < 20 AND B = 0 THEN POKE V + 16,
     PEEK ( V + 16 ) OR 1 : B = 1 : A = 64
75 IF A > 64 AND B THEN POKE V + 16,
     PEEK ( V + 16 ) AND 254 : B = 0 : A = 20
80 POKE V,A : A = PEEK ( V + 1 ) + ( K = 9 )
     * 8 - ( K = 23 ) * 8
85 IF A < 52 THEN A = 52
```

```
 90 IF A > 203 THEN A = 203
 95 POKE V + 1,A : IF MA > - 1 THEN 120
100 IF RND ( 1 ) > .1 THEN 500
101 POKE V + 23,0
105 MA = 0 : POKE V + 2,20 + RND ( 1 ) * 236 :
    POKE V + 16, PEEK ( V + 16 ) AND 253
110 IF RND ( 1 ) > .75 THEN POKE V + 16,
    PEEK ( V + 16 ) OR 2 : POKE V + 2,
    RND ( 1 ) * 64
115 POKE V + 3,52 : POKE V + 40,2 : POKE 2041,214 :
    POKE V + 21, PEEK ( V + 21 ) OR 2 :
    POKE 54290,17
120 ON MA GOTO 150,200,250,300,300,310
122 X = ( 600 - PEEK ( V + 3 ) ) * 25 :
    X1 = INT ( X / 256 ) : Y1 = X - X1 * 256
123 POKE 54287,X1 : POKE 54286,Y1
125 POKE V + 3, PEEK ( V + 3 ) + 8 :
    A = PEEK ( V + 3 ) : IF A < = 224 THEN 500
127 POKE 54276,0 : POKE 54273,5 : POKE 54272,103 :
    POKE 54277,68 : POKE 54276,129
130 X = PEEK ( V + 2 ) - 20 :
    IF PEEK ( V + 16 ) AND 2 THEN X = X + 256
132 X = INT ( X / 8 ) + 1944 : POKE X,6 :
    POKE X + 1,32 : POKE X + 2,7
135 POKE X + 40,8 : POKE X + 42,9 :
    POKE X + CO + 41,13 : POKE X + 41,10
140 MA = INT ( RND ( 1 ) * 3 + 1 ) : POKE V + 40,12
    : IF MA = 3 THEN POKE V + 40,6 :
    CH = INT ( RND ( 1 ) * 5 )
142 IF MA < 3 THEN POKE V + 23,2 : POKE V + 3,192 :
    GOTO 145
143 POKE V + 3,213
145 CR = X - 39 : RW = 0 : POKE 54290,0 : GOTO 500
150 POKE 2041,208 + PO : PO = 1 - PO :
    IF PO = 0 THEN 400
155 X = PEEK ( V + 2 ) - 8 :
    IF PEEK ( V + 16 ) AND 2 THEN X = X + 256
160 IF X < 20 THEN MA = 2 : X = 20 : GOTO 500
165 IF X > 255 THEN POKE V + 16,
    PEEK ( V + 16 ) OR 2 : POKE V + 2,X - 256 :
    GOTO 400
170 POKE V + 16, PEEK ( V + 16 ) AND 253 :
    POKE V + 2,X : GOTO 400
200 POKE 2041,210 + PO : PO = 1 - PO :
    IF PO = 0 THEN 400
205 X = PEEK ( V + 2 ) + 8 :
    IF PEEK ( V + 16 ) AND 2 THEN X = X + 256
210 IF X > 320 THEN MA = 1 : X = 320 : GOTO 500
215 IF X > 255 THEN POKE V + 16,
    PEEK ( V + 16 ) OR 2 : POKE V + 2,X - 256 :
    GOTO 400
220 POKE V + 16, PEEK ( V + 16 ) AND 253 :
    POKE V + 2,X : GOTO 400
250 IF PEEK ( V + 3 ) > 150 THEN POKE V + 3,
    PEEK ( V + 3 ) - 8 : POKE 2041,213 : GOTO 500
255 POKE 2041,212 + PO : PO = 1 - PO :
```

```
      X = PEEK ( V + 2 ) : IF PEEK ( V + 16 ) AND 2
      THEN X = X + 256
257 Y1 = PEEK ( V + 1 ) : X1 = PEEK ( V ) :
      IF PEEK ( V + 16 ) AND 1 THEN X1 = X1 + 256
260 Y = PEEK ( V + 3 ) : IF CH < 10 AND RND ( 1 )
      > .9 THEN CH = CH + 1
265 IF RND ( 1 ) < CH / 10 THEN X = X + ( X > X1 )
      * 8 - ( X < X1 ) * 8 : GOTO 275
270 X = X + INT ( RND ( 1 ) * 3 - 1 ) * 8 :
      IF X < 20 THEN X = 20
272 IF X > 320 THEN X = 320
275 IF RND ( 1 ) < CH / 10 THEN Y = Y + ( Y > Y1 )
      * 8 - ( Y < Y1 ) * 8 : GOTO 285
280 Y = Y + INT ( RND ( 1 ) * 3 - 1 ) * 8 :
      IF Y < 52 THEN Y = 52
285 POKE V + 3,Y : IF X > 255 THEN POKE V + 16,
      PEEK ( V + 16 ) OR 2 : POKE V + 2,X - 256 :
      GOTO 500
290 POKE V + 16, PEEK ( V + 16 ) AND 253 :
      POKE V + 2,X : GOTO 500
300 POKE 2041,218 : POKE V + 40,2 : MA = MA + 1 :
      GOTO 500
310 POKE V + 21, PEEK ( V + 21 ) AND 253 : MA = - 1
      : GOTO 500
400 IF SH < > 0 OR RND ( 1 ) < .1 THEN 500
402 POKE 54276,0 : POKE 54273,72 : POKE 54272,169 :
      POKE 54277,79 : POKE 54276,129
405 SH = - 1 : X = PEEK ( V + 2 ) - 18 :
      IF MA = 2 THEN SH = 1 : X = PEEK ( V + 2 )+18
406 IF PEEK ( V + 16 ) AND 2 THEN X = X + 256
407 IF X > 255 THEN POKE V + 16,
      PEEK ( V + 16 ) OR 4 : POKE V + 4,X - 256 :
      GOTO 410
408 POKE V + 16, PEEK ( V + 16 ) AND 251 :
      POKE V + 4,X
410 POKE V + 5, PEEK ( V + 3 ) : POKE V + 41,8 :
      POKE V + 21, PEEK ( V + 21 ) OR 4
500 IF PEEK ( V + 30 ) AND 1 THEN 1000
505 IF SH = 0 THEN 530
510 X = PEEK ( V + 4 ) + SH * 16 :
      IF PEEK ( V + 16 ) AND 4 THEN X = X + 256
515 IF X < 20 OR X > 320 THEN POKE V + 21,
      PEEK ( V + 21 ) AND 251 : SH = 0 : GOTO 530
520 IF X > 255 THEN POKE V + 16, PEEK ( V + 16 )
      OR 4 : POKE V + 4,X - 256 : GOTO 530
525 POKE V + 16, PEEK ( V + 16 ) AND 251 :
      POKE V + 4,X : I = PEEK ( V + 30 )
530 IF RW > 0 THEN 550
535 IF RND ( 1 ) < .95 OR MA < 1 THEN 600
540 RW = CR
550 POKE RW + CO,2 : POKE RW,3 : R1 = RW + INT
      ( RND ( 1 ) * 3 - 1 )
555 IF R1 < 1904 OR R1 > 1943 THEN RW = 0 :
      GOTO 600
560 POKE R1 + CO,2 : IF R1 < RW THEN POKE R1,5 :
      GOTO 575
```

```
565 IF R1 > RW THEN POKE R1,4 : GOTO 575
570 POKE R1,3
575 RW = R1
600 IF PS < > 0 OR K < > 13 THEN 650
602 POKE 54276,0 : POKE 54273,57 : POKE 54272,172 :
      POKE 54277,79 : POKE 54276,129
605 X = PEEK ( V ) + 24 : Y = PEEK ( V + 1 ) + 13 :
      IF PEEK ( V + 16 ) AND 1 THEN X = X + 256
610 PS = 1 : IF DI THEN PS = - 1 : X = X - 48
615 POKE V + 7,Y : IF X > 255 THEN
      POKE V + 6,X - 256 : POKE V + 16,
      PEEK ( V + 16 ) OR 8 : GOTO 625
620 POKE V + 6,X : POKE V + 16,
      PEEK ( V + 16 ) AND 247
625 POKE V + 42,8 : POKE 2043,215 : POKE V + 21,
      PEEK ( V + 21 ) OR 8
650 IF PS = 0 THEN 675
655 X = PEEK ( V + 6 ) + PS * 12 :
      Y = PEEK ( V + 7 ) : IF PEEK ( V + 16 ) AND 8
      THEN X = X + 256
660 IF X < 20 OR X > 320 THEN POKE V + 21,
      PEEK ( V + 21 ) AND 247 : PS = 0 : GOTO 675
665 IF X > 255 THEN POKE V + 6,X - 256 :
      POKE V + 16, PEEK ( V + 16 ) OR 8 : GOTO 675
670 POKE V + 6,X : POKE V + 16,
      PEEK ( V + 16 ) AND 247
675 IF PEEK ( V + 30 ) AND 2 THEN 677
676 GOTO 690
677 SC = SC + 10 : MA = 4 : POKE V + 21,
      PEEK ( V + 21 ) AND 247 : PS = 0
680 POKE 54276,0 : POKE 54273,34 : POKE 54272,75 :
      POKE 54277,74 : POKE 54276,129
685 POKE 54290,0
690 GOTO 55
1000 POKE 2040,218 : SH = 0 : POKE V + 21,
      PEEK ( V + 21 ) AND 251 : POKE 54276,0 :
      POKE 54277,143
1005 POKE 54273,3 : POKE 54272,155 : POKE 54276,129
      : FOR I = 1 TO 2000 : NEXT
1010 LI = LI - 1 : I = PEEK ( V + 30 ) :
      POKE V + 21, PEEK ( V + 21 ) AND 254 :
      POKE V,20 : POKE V + 1,99
1012 POKE V + 16, PEEK ( V + 16 ) AND 254 : DI = 0
1015 IF LI > 0 THEN POKE 2040,16 : POKE V + 21,
      PEEK ( V + 21 ) OR 1 : I = PEEK ( V + 30 ) :
      GOTO 55
1017 POKE 54273,0 : POKE 54272,0 : POKE 54287,0 :
      POKE 54286,0
1020 POKE V + 24,20 : POKE V + 21,0 : PRINT "♥ ◙
      HARD ▲LINES"
1025 PRINT "Ⓠ Ⓠ Ⓠ Ⓠ ⇑YOU ▲SCORED";SC
1030 PRINT "Ⓠ Ⓠ ▣THE ▲MARTIANS ▲WILL ▲OVER ▲RUN
       ▲EARTH"
1035 PRINT "Ⓠ Ⓠ Ⓠ ▒ANOTHER ▲GAME?"
1040 GET A$ : IF A$ = "Y" THEN POKE V + 24,28 :
      PRINT "♥"; : GOTO 15
```

```
1045 IF A$ < > "N" THEN 1040
1050 POKE V + 22, PEEK ( V + 22 ) AND 239
1060 END
10000 DATA 3,192,0,0,64,0,1,64,0,1,80,0,3,84,0
      ,15,85,64,5,85,80,5,85,80,1,85,64
10005 DATA 1,85,0,0,252,0,0,136,0,2,10,0,8,0,128
      ,32,0,32,32,0,8,32,0,8,32,0,8
10010 DATA 32,0,8,32,0,8,136,0,34,99
10015 DATA 60,0,0,4,0,0,20,0,0,21,0,0,53,64,0,245
      ,84,0,85,85,0,85,85,0
10020 DATA 21,84,0,21,80,0,15,192,0,8,128,0,8,128
      ,0,32,32,0,32,32,0,128,8,0
10025 DATA 128,8,0,32,8,0,32,8,0,8,8,0,34,34,0,99
10030 DATA 0,3,192,0,1,0,0,1,64,0,5,64,0,21,192
      ,1,85,240,5,85,80,5,85,80,1,85,64
10035 DATA 0,85,64,0,63,0,0,34,0,0,160,128,2,0
      ,32,8,0,8,32,0,8,32,0,8,32,0,8
10040 DATA 32,0,8,32,0,8,136,0,34,99
10045 DATA 0,0,60,0,0,16,0,0,20,0,0,84,0,1,92,0
      ,21,95,0,85,85,0,85,85,0,21,84
10050 DATA 0,5,84,0,3,240,0,2,32,0,2,32,0,8,8,0
      ,8,8,0,32,2,0,32,2,0,32,8,0,32,8
10055 DATA 0,32,32,0,136,136,99
10060 DATA 0,0,0,0,0,0,0,0,0,64,0,4,20,0,80,1
      ,1,0,0,68,0,0,32,0,0,32,0,0,168,0
10065 DATA 2,102,0,2,170,0,3,239,0,2,238,0,0
      ,168,0,0,168,0,0,100,0,1,205,0
10070 DATA 4,0,64,4,0,64,17,1,16,99
10075 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1
      ,0,4,68,64,16,32,16,64,32,4,0,168,0
10080 DATA 2,102,0,2,170,0,3,239,0,2,238,0,0
      ,168,0,0,168,0,0,100,0,1,205,0
10085 DATA 4,0,64,4,0,64,17,1,16,99
10090 DATA 0,0,0,0,0,0,0,0,0,192,32,12,48,32
      ,48,12,32,192,3,35,0,0,84,0,0,84,0
10095 DATA 170,86,168,170,86,168,0,84,0,0,84
      ,0,3,35,0,12,32,192,48,32,48
10100 DATA 192,32,12,0,0,0,0,0,0,0,0,0,0,0,0,99
10105 DATA 255,204,255,0,0,0,0,0,0,0,0,0,0,0
      ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
10110 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
      ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,99
10115 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
      ,0,0,0,0,0,0,0,0,0,0,12,0,0,15,0,0
10120 DATA 15,195,192,15,255,240,42,170,170
      ,2,250,168,0,254,0,0,252,0,3,240,0
10125 DATA 3,192,0,3,0,0,0,0,0,99
10130 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
      ,0,0,0,0,0,0,0,0,0,0,0,48,0,0,240
10135 DATA 3,195,240,15,255,240,170,170,168,42
      ,175,128,0,63,0,0,63,0,0,15,192
10140 DATA 0,3,192,0,0,192,0,0,0,99
10145 DATA 0,0,0,0,8,0,2,8,8,0,8,32,0,42,128
      ,0,174,160,2,239,128,42,255,160
10150 DATA 11,255,224,2,255,224,2,255,128,2
      ,190,0,8,186,128,0,184,32,0,40,0
```

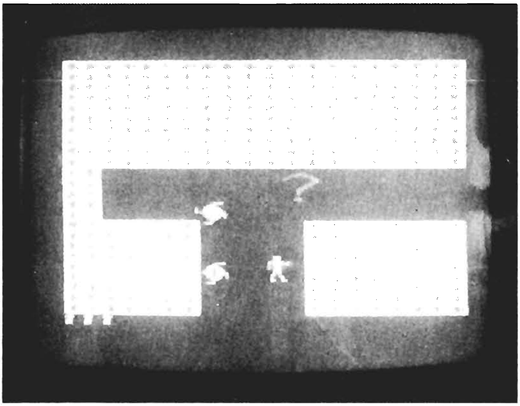
```
10155 DATA 0,32,0,32,32,0,0,0,128,2,0,2,0,0,0,0,0,0,99
10160 DATA 255,195,195,195,195,195,195,255,60
      ,60,24,126,24,24,102,102
10165 DATA 255,255,255,255,255,255,255,255,12
      ,51,192,48,12,51,51,192
10170 DATA 0,0,192,48,204,6,3,0,0,0,0,3,12,48,207,0
10175 DATA 0,0,0,128,128,192,224,240,0,0,0,0,1
      ,1,1,3,240,248,252,252,252,252
10180 DATA 254,255,3,3,7,7,15,31,127,255,0
      ,105,85,85,85,85,85,255
```

CHEXSUM

0=4223	120=1950	290=2918
1=595	122=3947	300=2791
2=3411	123=1344	310=2890
5=3647	125=3549	400=1856
10=4656	127=3331	402=3378
15=5252	130=3214	405=4348
20=4931	132=3088	406=1905
25=8416	135=3241	407=3956
30=6312	140=4996	408=2263
35=7096	142=2760	410=3447
40=8041	143=676	500=1444
45=1905	145=2466	505=870
50=4595	150=2762	510=3607
52=5663	155=3158	515=4125
53=3332	160=2165	520=3956
55=4534	165=3951	525=3311
56=1056	170=2917	530=882
57=1063	200=2763	535=1739
58=904	205=3158	540=507
60=3398	210=2260	550=3148
65=3575	215=3951	555=2687
70=3571	220=2917	560=2683
75=3413	250=4023	565=1818
80=3103	255=4888	570=383
85=1036	257=3944	575=488
90=1154	260=3383	600=1824
95=1691	265=4201	602=3384
100=1111	270=3040	605=4352
101=618	272=1192	610=2250
105=3652	275=4210	615=4736
110=3799	280=3048	620=2310
115=4324	285=4744	625=2931

```
  650=885        10080=3711
  655=4636       10085=1425
  660=4141       10090=4157
  665=3966       10095=3561
  670=2310       10100=2118
  675=1384       10105=4353
  676=538        10110=3768
  677=3514       10115=4092
  680=3331       10120=4058
  685=533        10125=1326
  690=477        10130=4205
 1000=4191       10135=4085
 1005=3207       10140=1453
 1010=4756       10145=3834
 1012=2051       10150=3818
 1015=4538       10155=2477
 1017=2393       10160=3386
 1020=2746       10165=3301
 1025=1409       10170=2360
 1030=3018       10175=3922
 1035=1478       10180=3121
 1040=2971
 1045=1182     TOTAL= 480589
 1050=1579
 1060=129
10000=4319
10005=4065
10010=1477
10015=3407
10020=3805
10025=2246
10030=4426
10035=3933
10040=1477
10045=4041
10050=4279
10055=1219
10060=4112
10065=3711
10070=1425
10075=4252
```

Castle of Doom



Explore the rooms within the castle walls, look for treasure and try to avoid the slime monsters and killer robots.

Points are gained by moving over question marks and killing monsters or robots with your Proton gun.

As you move off the screen from one room you are presented with another. Move back and you return to the same room but now with different contents!

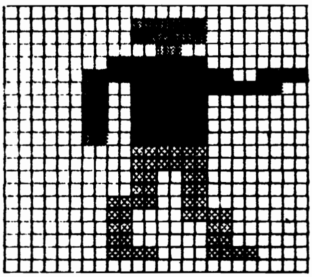
You are killed if you touch the electrified walls or any of the monsters. Some monsters are faster than others and the robots are very dangerous.

Controls are:

'W' up 'D' right
'X' down 'A' left
'S' to fire your gun

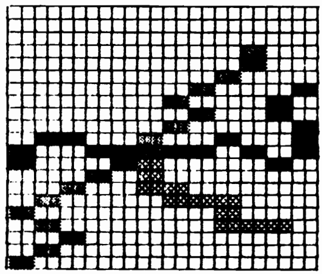
PROGRAM STRUCTURE	LINES
INITIALIZATION	0 — 50
READ KEY PRESSED, GET POSITION OF PLAYER	55
MOVE PLAYER HORIZONTALLY, CHECKING FOR DOORS	56 — 70
MOVE PLAYER VERTICALLY, CHECKING FOR DOORS	75 — 80
CHECK IF PLAYER HIT WALL OR MONSTER OR QUESTION MARK	80 — 110
INCREASE SCORE IF HIT QUESTION MARK	115 — 125
HAS PLAYER FIRED?	150
SET UP SHOT	151 — 162
MOVE SHOT UNTIL HITS SOMETHING OR DISAPPEARS. IF HIT WALL EXIT, IF HIT MONSTER INCREASE SCORE AND EXIT	170 — 198
TURN OFF SHOT SOUNDS	199
IF MONSTER ONE EXISTS THEN MOVE IT	200 — 290
IF MONSTER TWO EXISTS THEN MOVE IT	300 — 390
TO MAIN LOOP AGAIN	400
SUBROUTINE TO GENERATE NEW ROOM, CREATE TWO NEW MONSTERS AND A QUESTION MARK (RANDOMLY)	1000 — 1200
PLAYER KILLED, IF NO. OF LIVES LEFT EQUALS 0 THEN END GAME. ASK PLAYER IF WANT TO PLAY AGAIN	2000 — 2025
DATA	1000 — 12120

VARIABLES
V = VIDEO CHIP
R = ROOM NUMBER
L1 = LIVES LEFT
SC = SCORE
KB = PLACE TO READ KEY PRESSED FROM
K = KEY PRESSED
I, A, B, U, X, Y, YX, YY, B$, DS = TEMPORARY STORAGE
S1, S2 = OFFSET TO MOVE MONSTERS
DI = DIRECTION OF MOVEMENT
B = HIGH X — BIT FOR SPRITES
SH = SHOT IN PROGRESS
M1 = TYPE OF MONSTER 1
M2 = TYPE OF MONSTER 2
A$ = CONTAINS ROOM DATA



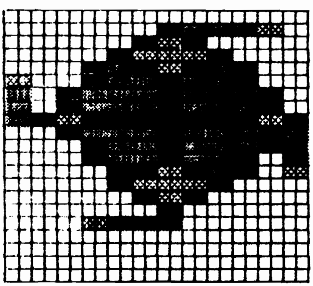
SPRITE

PLAYER



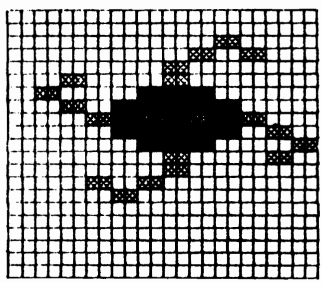
SPRITE

CREATURE 1



SPRITE

CREATURE 2



SPRITE

CREATURE 3

CASTLE OF DOOM

```
  0 POKE 55,255 : POKE 56,47 : V = 53248 :
      FOR I = 12544 TO 12551 : POKE I,0 : NEXT :
      POKE V + 32,0
  1 POKE V + 33,0 : FOR I = 12288 TO 12319 : READ A
      : POKE I,A : NEXT : POKE 54296,15
  2 POKE V + 24,28 : POKE V + 22, PEEK ( V + 22 )
      OR 16 : PRINT "♥" : FOR I = 12352 TO 12991 :
      READ A : POKE I,A
  3 NEXT : POKE V + 37,10 : POKE V + 38,5
  5 DIM A$ ( 24 ) : R = 0 : POKE 54278,0 :
      POKE 54292,0
 10 FOR I = 0 TO 24 : READ A$ ( I ) : NEXT :
      LI = 3 : KB = 197 : SC = 0
 15 POKE V + 16,0 : POKE V,150 : POKE V + 1,120 :
      POKE V + 39,6 : POKE V + 28,15 :
      POKE 2040,193 : DI = 0
 20 POKE V + 40,2
 50 GOSUB 1000 : SH = 0
 55 K = PEEK ( KB ) : A = PEEK ( V ) + ( K = 10 )
      * 8 - ( K = 18 ) * 8 : B = PEEK ( V + 16 ) AND 1
 56 IF K = 10 THEN DI = 1
 57 IF K = 18 THEN DI = 0
 58 POKE 2040,193 + DI
 60 IF A < 0 AND B THEN B = 0 : A = 252
 65 IF A > 255 AND B = 0 THEN B = 1 : A = 0
 67 IF B AND A > 60 THEN POKE V + 16,
      PEEK ( V + 16 ) AND 254 : POKE V,20 :
      R = R + 1 : GOTO 50
 68 IF B = 0 AND A < 20 THEN POKE V + 16,
      PEEK ( V + 16 ) OR 1 : POKE V,60 : R = R - 1
      : GOTO 50
 70 POKE V,A : POKE V + 16, PEEK ( V + 16 ) OR B :
      IF B = 0 THEN POKE V + 16, PEEK ( V + 16 )
      AND 254
 75 A = PEEK ( V + 1 ) + ( K = 9 ) * 8 -
      ( K = 23 ) * 8
 77 IF A < 52 THEN POKE V + 1,220 : R = R + 5 :
      GOTO 50
 78 IF A > 220 THEN POKE V + 1,52 : R = R - 5 :
      GOTO 50
 80 POKE V + 1,A : IF PEEK ( V + 31 ) AND 1
      THEN 2000
 85 IF PEEK ( V + 30 ) AND 1 THEN 95
 90 GOTO 150
 95 IF PEEK ( V + 30 ) > 128 THEN 115
100 IF PEEK ( V + 30 ) AND 8 THEN 2000
110 IF PEEK ( V + 30 ) AND 4 THEN 2000
112 GOTO 150
115 SC = SC + INT ( RND ( 1 ) * 5 + 1 ) * 50
125 POKE V + 21, PEEK ( V + 21 ) AND 127 : T = - 1
150 IF K < > 13 THEN 200
151 POKE  54276,0 : POKE 54277,204 : POKE 54273,137
      : POKE 54272,43 : POKE 54276,33
152 POKE 54290,0 : POKE 54291,136 : POKE 54287,69 :
```

```
      POKE V + 16, PEEK ( V + 16 ) AND 253
153 POKE 54286,149 : DS = 8 : SH = 1 :
      X = PEEK ( V ) + 24 : IF PEEK ( V + 16 )
      AND 1 THEN X = X + 256
154 POKE 54290,17 : IF DI THEN DS = - 8 : X = X - 48
155 B = 0 : IF X > 256 THEN B = 2 : X = X - 256
160 POKE V + 2,X : POKE V + 3, PEEK ( V + 1 ) :
      POKE V + 16, PEEK ( V + 16 ) OR B :
      POKE 2041,195 + DI
162 POKE V + 21, PEEK ( V + 21 ) OR 2
170 A = PEEK ( V + 2 ) + DS : B = PEEK ( V + 16 )
      AND 2 : IF A < 0 AND B THEN A = 248 : B = 0
175 IF A > 255 AND B = 0 THEN B = 2 : A = 0
180 POKE V + 2,A : POKE V + 16, PEEK ( V + 16 ) OR B
      : IF B = 0 THEN POKE V + 16, PEEK ( V + 16 )
      AND 253
185 IF PEEK ( V + 31 ) AND 2 THEN POKE V + 21,
      PEEK ( V + 21 ) AND 253 : GOTO 199
190 IF ( A < 20 AND B = 0 ) OR ( A > 60 AND B )
      THEN POKE V + 21, PEEK ( V + 21 ) AND 253 :
      GOTO 200
191 U = PEEK ( V + 30 ) : IF ( U AND 128 ) THEN
      IF ( U AND 2 ) THEN 198
192 IF U AND 2 THEN 194
193 GOTO 198
194 IF ( U AND 4 ) AND M1 < 5 THEN POKE V + 21,
      PEEK ( V + 21 ) AND 253 : SH = 0 :
      SC = SC + M1 * 10 : M1 = 5 : GOTO 199
195 IF ( U AND 8 ) AND M2 < 5 THEN POKE V + 21,
      PEEK ( V + 21 ) AND 253 : SH = 0 :
      SC = SC + M2 * 10 : M2 = 5 : GOTO 199
198 GOTO 170
199 POKE 54276,0 : POKE 54277,72 : POKE 54273,4 :
      POKE 54272,73 : POKE 54276,129
200 IF M1 = 0 THEN 300
205 X = PEEK ( V + 4 ) : Y = PEEK ( V + 5 ) :
      IF PEEK ( V + 16 ) AND 4 THEN X = X + 256
210 YX = PEEK ( V ) : YY = PEEK ( V + 1 ) :
      IF PEEK ( V + 16 ) AND 1 THEN YX = YX + 256
212 IF M1 > 4 THEN 275
215 IF M1 = 4 THEN 250
220 IF M1 = 3 AND RND ( 1 ) > .3 THEN 250
225 X1 = X + ( X > YX ) * S1 - ( X < YX ) * S1 :
      Y1 = Y + ( Y > YY ) * S1 - ( Y < YY ) * S1
227 IF X1 > 255 THEN X1 = X1 - 256 : POKE V + 16,
      PEEK ( V + 16 ) OR 4 : GOTO 235
230 POKE V + 16, PEEK ( V + 16 ) AND 251
235 POKE V + 4,X1 : POKE V + 5,Y1
247 GOTO 300
250 X1 = X + INT ( RND ( 1 ) * 3 - 1 ) * S1 :
      Y1 = Y + INT ( RND ( 1 ) * 3 - 1 ) * S1
251 IF X1 < 20 OR X1 > 320 OR Y1 < 52 OR Y1 > 220
      THEN X1 = X : Y1 = Y
255 POKE V + 16, PEEK ( V + 16 ) AND 251 :
      IF X1 > 255 THEN X1 = X1 - 256 : POKE V + 16,
      PEEK ( V + 16 ) OR 4
```

```
 260 POKE V + 4,X1 : POKE V + 5,Y1
 270 GOTO 300
 275 IF M1 = 10 THEN 290
 280 POKE V + 41,1 : POKE V + 28, PEEK ( V + 28 )
       AND 251 : M1 = M1 + 1 : GOTO 300
 290 POKE V + 28, PEEK ( V + 28 ) OR 4 : M1 = 0 :
       POKE V + 21, PEEK ( V + 21 ) AND 251
 300 IF M2 = 0 THEN 400
 305 X = PEEK ( V + 6 ) : Y = PEEK ( V + 7 ) :
       IF PEEK ( V + 16 ) AND 8 THEN X = X + 256
 310 YX = PEEK ( V ) : YY = PEEK ( V + 1 ) :
       IF PEEK ( V + 16 ) AND 1 THEN YX = YX + 256
 312 IF M2 > 4 THEN 375
 315 IF M2 = 4 THEN 350
 320 IF M2 = 3 AND RND ( 1 ) > .3 THEN 350
 325 X1 = X + ( X > YX ) * S2 - ( X < YX ) * S2 :
       Y1 = Y + ( Y > YY ) * S2 - ( Y < YY ) * S2
 327 IF X1 > 255 THEN X1 = X1 - 256 : POKE V + 16,
       PEEK ( V + 16 ) OR 8 : GOTO 335
 330 POKE V + 16, PEEK ( V + 16 ) AND 247
 335 POKE V + 6,X1 : POKE V + 7,Y1
 347 GOTO 400
 350 X1 = X + INT ( RND ( 1 ) * 3 - 1 ) * S2 :
       Y1 = Y + INT ( RND ( 1 ) * 3 - 1 ) * S2
 351 IF X1 < 20 OR X1 > 320 OR Y1 < 52 OR Y1 > 220
       THEN X1 = X : Y1 = Y
 355 POKE V + 16, PEEK ( V + 16 ) AND 247 :
       IF X1 > 255 THEN X1 = X1 - 256 : POKE V + 16,
       PEEK ( V + 16 ) OR 8
 360 POKE V + 6,X1 : POKE V + 7,Y1
 370 GOTO 400
 375 IF M2 = 10 THEN 390
 380 POKE V + 42,1 : POKE V + 28, PEEK ( V + 28 )
       AND 247 : M2 = M2 + 1 : GOTO 400
 390 POKE V + 28, PEEK ( V + 28 ) OR 8 : M2 = 0 :
       POKE V + 21, PEEK ( V + 21 ) AND 247
 400 GOTO 55
1000 A$ = A$ ( R ) : B$ = "0000" : PRINT "♥" :
       POKE V + 35,8 + INT ( R / 5 ) : POKE V + 21,0
       : FOR I = 1 TO 20
1005 IF LEN ( A$ ) > = 4 THEN B$ = LEFT$ ( A$,4 ) :
       A$ = RIGHT$ ( A$, LEN ( A$ ) - 4 )
1010 PRINT "◘"; TAB( ( I - 1 ) * 2 ) ; :
       D = VAL ( LEFT$ ( B$,2 ) ) :
       U = VAL ( RIGHT$ ( B$,2 ) )
1015 IF D = 0 THEN 1025
1020 FOR J = 1 TO D : PRINT "ƏA▣ ▮▮ ▮▮"; : NEXT
1025 PRINT "◘▣▣▣▣▣▣▣▣▣▣▣▣▣▣
       ▣▣▣▣▣▣▣▣▣"; TAB( ( I - 1 ) * 2 ) ;
1030 IF U > 0 THEN FOR J = 1 TO U : PRINT "ƏA□ ▮▮ ▮▮"
       ; : NEXT
1035 NEXT I : T = - 1 : IF RND ( 1 ) < .6 THEN 1065
1037 T = 0 : POKE V + 14,100 + RND ( 1 ) * 150
1040 POKE V + 15,100 + RND ( 1 ) * 50 : POKE V + 16,
       PEEK ( V + 16 ) AND 127 : POKE V + 46,2
1045 POKE V + 29, PEEK ( V + 29 ) OR 128 :
```

```
        POKE 2047,197 : IF KE THEN 1055
1055 POKE V + 21,128 : I = PEEK ( V + 31 ) +
        PEEK ( V + 31 ) + PEEK ( V + 31 )
1060 IF PEEK ( V + 31 ) AND 128 THEN T = - 1 :
        POKE V + 21,0
1065 POKE V + 21, PEEK ( V + 21 ) OR 1 :
        I = PEEK ( V + 31 ) : I = PEEK ( V + 30 )
1070 M1 = 0 : M2 = 0 : IF RND ( 1 ) < .3 THEN 1110
1072 POKE V + 4, RND ( 1 ) * 236 + 20 : POKE V + 5,
        RND ( 1 ) * 168 + 52
1075 M1 = INT ( RND ( 1 ) * 4 + 1 ) :
        POKE 2042,198 + M1 : POKE V + 41,6 : S1 = 2
1080 IF M1 = 2 THEN POKE V + 41,11 : S1 = 6
1085 IF M1 = 3 THEN POKE V + 41,9 : S1 = 4
1090 IF M1 = 4 THEN POKE V + 41,14 : S1 = 4
1100 POKE V + 21, PEEK ( V + 21 ) OR 4 :
        I = PEEK ( V + 31 )
1110 IF RND ( 1 ) < .3 THEN 1150
1112 POKE V + 6, RND ( 1 ) * 236 + 20 :
        POKE V + 7, RND ( 1 ) * 168 + 52
1115 M2 = INT ( RND ( 1 ) * 4 + 1 ) :
        POKE 2043,198 + M2 : POKE V + 42,6 : S2 = 2
1120 IF M2 = 2 THEN POKE V + 42,11 : S2 = 6
1125 IF M2 = 3 THEN POKE V + 42,9 : S2 = 4
1130 IF M2 = 4 THEN POKE V + 42,14 : S2 = 4
1135 POKE V + 21, PEEK ( V + 21 ) OR 8 :
        I = PEEK ( V + 31 )
1150 PRINT "■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■
        ■ ■ ■ ■ ■ ■ ■ ■ ■"; : FOR I = 1 TO LI :
        PRINT "■B ▲ "; : NEXT
1155 POKE V + 28,15
1200 RETURN
2000 POKE V + 28, PEEK ( V + 28 ) AND 254 :
        POKE 54276,0 : POKE 54277,143 : POKE 54273,34
2001 POKE 54272,75 : POKE 54276,129 : POKE V + 39,1 :
        FOR I = 1 TO 2000 : NEXT
2004 POKE V,150 : POKE V + 1,120
2005 POKE V + 28, PEEK ( V + 28 ) OR 1 :
        POKE V + 39,6 : POKE V + 21,0 : LI = LI - 1 :
        R = 0 : IF LI > 0 THEN 50
2010 POKE V + 24,20 : PRINT "■ ■HARD ▲ LINES, ▲ YOU
         ▲ WERE ▲ KILLED"
2012 PRINT "■ ■ ■ ■YOU ▲ SCORED"SC
2015 PRINT "■ ■ ■ANOTHER ▲ GAME ▲ ?"
2017 POKE 54273,0 : POKE 54272,0 : POKE 54287,0 :
        POKE 54286,0
2020 GET A$ : IF A$ = "Y" THEN SC = 0 : LI = 3 :
        POKE V + 24,28 : GOTO 15
2025 IF A$ < > "N" THEN 2020
2030 END
10000 DATA 255,234,234,234,234,234,234,255,255
        ,171,171,171,171,171,171,255
10005 DATA 24,24,255,24,60,108,108,108,0,6,249,70,0
        ,0,0,0
11000 DATA 0,0,0,0,21,0,0,21,0,0,4,0,0,170,0,2
        ,170,138,2,42,164,2,42,0,2,42,0
```

```
11005 DATA 2,42,0,1,42,0,0,63,0,0,63,0,0,51,0
      ,0,51,0,0,243,0,0,195,192,0,192,192
11010 DATA 0,192,192,0,240,240,0,0,0,99
11015 DATA 0,0,0,0,84,0,0,84,0,0,16,0,0,170,0
      ,162,170,128,26,168,128,0,168,128
11020 DATA 0,168,128,0,168,128,0,168,64,0,252
      ,0,0,252,0,0,204,0,0,204,0,0,207,0
11025 DATA 3,195,0,3,3,0,3,3,0,15,15,0,0,0,0,99
11030 DATA 0,0,0,0,0,0,3,0,0,0,32,0,12,168,0
      ,194,168,0,12,168,0,0,32,0,3,0,0,0,0
11035 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
      ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
11040 DATA 99,0,0,0,0,0,0,0,0,192,0,8,0,0,42
      ,48,0,42,131,0,42,48,0,8,0,0,0,192
11045 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
      ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
11050 DATA 0,0,99,0,56,0,0,126,0,1,231,128,3,
      1,192,6,0,224,0,0,224,0,0,224
11055 DATA 0,1,192,0,3,128,0,7,0,0,14,0,0,28
      ,0,0,56,0,0,48,0,0,48,0,0,48,0,0,0,0
11060 DATA 0,0,0,0,48,0,0,120,0,0,48,0,99,0
      ,0,0,0,0,0,0,0,0,0,0,0,0,48,0,0,206
11065 DATA 0,3,3,15,252,3,3,3,3,3,0,206,0,0
      ,48,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
11070 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,99
11075 DATA 0,0,0,0,0,0,0,0,192,0,3,48,0,12,0
      ,12,12,0,48,42,0,12,170,128
11080 DATA 3,153,176,0,170,140,0,42,3,0,12
      ,12,0,12,0,3,48,0,0,192,0,0,0,0,0,0,0
11085 DATA 0,0,0,0,0,0,0,0,0,0,0,99,0,8,0
      ,0,42,0,0,38,0,0,42,0,0,8,0,2,170,160
11090 DATA 2,46,32,2,46,32,3,42,48,2,21,32
      ,2,42,32,0,42,0,0,34,0,0,128,128
11095 DATA 0,192,192,0,128,128,0,128,128,0
      ,34,0,0,34,0,0,162,128,2,162,160,99
11100 DATA 0,0,0,0,9,92,0,46,0,0,191,128,2
      ,110,96,194,89,96,73,89,88,73,89,88
11105 DATA 110,170,174,9,89,89,2,89,89,2,89
      ,97,0,174,99,0,191,128,0,46,0
11110 DATA 0,8,0,3,84,0,0,0,0,0,0,0,0,0,0,0,99
11115 DATA 0,0,0,0,0,0,0,0,0,0,0,16,0,0,16,0
      ,0,64,0,1,2,0,4,8,0,1,8,0,4,2
11120 DATA 40,16,130,130,154,34,128,112,8,1
      ,48,0,4,60,0,16,15,192,64,0,192
11125 DATA 16,0,252,4,0,0,16,0,0,64,0,0,99
12000 DATA 240024000504050405040304030400040000
      40004000400041009
12005 DATA 1009100910090504050405040504051500040000
      400040004100910091004100410009
12010 DATA 1009
12015 DATA 1009100910091009100910091009100091000
      9000900090009000900024
12020 DATA 0024002400240504050405040504050404000
      900090009000900090504050405040024
12025 DATA 2400240010091009100910091009100091000
      10001000100010001009
```

```
12030 DATA 1009100910091009100910091009100
      900000000000000000024
12035 DATA 24002400050405040504050405040504000
      4000400040004050405041009
12040 DATA 10091009100910091009100910091009
      10001000100010001009
12045 DATA 10091009050405040504050405040504000000
      000000000003020302030203020024
12050 DATA 24002400240024002400240024002
      40000090009000900091009
12055 DATA 1009100910091009100910091009100
      910001000100010000024
12060 DATA 2400240003020302030203020302030200000000
      000000000302030203021009
12065 DATA 10091009100910091004100410041004000
      400040004000410041004100410041009
12070 DATA 1009100910091009100910091009100900
      000000000000000024
12075 DATA 240024002400240005040504050405040000000
      00000000005040504050405040024
12080 DATA 24002400240024002400240024002400000
      90009000900091009
12085 DATA 100910091009100910091009100910091000
      1000100010000024
12090 DATA 2400240003020302030203020302030200000
      0000000000003020302030203021009
12095 DATA 1009100910091009100910091009100900
      000000000000000024
12100 DATA 240024000504050405040504050405040
      50005000500050005040504050405041009
12105 DATA 10091009100910091009100910091009
      10001000100010001009
12110 DATA 1009100910091009050405040504050405
      0405040504050405040024
12115 DATA 24002400240024002400240024002400
      10001000100010001009
12120 DATA 10090302030203020302030215020302030
      003000300030003020302030203020024
```

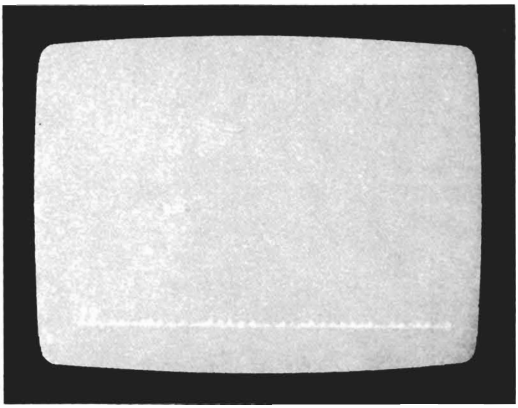
CHEXSUM

0=4275
1=3523
2=4979
3=1592
5=2087
10=2985
15=4706
20=614
50=814
55=4568
56=1056
57=1063
58=900
60=1670
65=1921
67=4360
68=4509
70=4371
75=2592
77=2602
78=2598
80=2099
85=1319
90=526
95=1505
100=1444
110=1448
112=526
115=2031
125=2146
150=1022
151=3424
152=3717
153=4823
154=2447
155=2233
160=4585
162=1483
170=4330
175=1920
180=4610
185=3526
190=4501
191=2821
192=800
193=542
194=5910
195=5922
198=528
199=3278
200=839
205=3951
210=4115
212=849
215=844
220=1648
225=5380
227=3890
230=1572
235=1400
247=525
250=4133
251=3737
255=5004
260=1400
270=525
275=898
280=3740
290=3638
300=839
305=3953
310=4115
312=849
315=844
320=1648
325=5382
327=3894
330=1577
335=1404
347=529
350=4133
351=3737
355=5013
360=1404
370=529
375=899
380=3746
390=3637
400=477
1000=4913
1005=3619
1010=3696
1015=836
1020=1734
1025=1825
1030=2568
1035=1979
1037=2012
1040=4013
1045=3047
1055=3551
1060=2549
1065=3558
1070=2034
1072=3122
1075=3573
1080=1806
1085=1754
1090=1803
1100=2525
1110=1171
1112=3126
1115=3574
1120=1808
1125=1755
1130=1805
1135=2525
1150=2682
1155=685
1200=143
2000=3626
2001=3219
2004=1220

```
2005=5251
2010=3535
2012=1438
2015=1501
2017=2393
2020=3383
2025=1183
2030=129
10000=3726
10005=2660
11000=4018
11005=4394
11010=1695
11015=4201
11020=4294
11025=2119
11030=4376
11035=4088
11040=4169
11045=4088
11050=3712
11055=4408
11060=4334
11065=4146
11070=1783
11075=3436
11080=4272
11085=4359
11090=3705
11095=4066
11100=4095
11105=3587
11110=2262
11115=3557
11120=3718
11125=1881
12000=3042
12005=4338
12010=346
12015=3293
12020=4341
12025=3040
12030=3043
12035=3506
12040=3054
12045=4317
12050=3042
12055=3047
12060=3842
12065=4334
12070=3043
12075=4315
12080=3042
12085=3047
12090=4306
12095=3043
12100=4334
12105=3054
12110=3285
12115=3030
12120=4311

TOTAL= 519162
```

U.F.O.



Another U.F.O. game with aliens attacking your craft. This one however contains a Machine Code routine to scroll the screen.

Shoot down the attacking U.F.O.'s with one of your missiles.

The game ends if you crash or if a U.F.O. gets past you.

Your controls are:
- 'W' up
- 'X' down
- 'S' to fire

PROGRAM STRUCTURE	LINES
INITIALIZE SPRITE	0 — 17
PRINT GROUND	20
MACHINE CODE ROUTINE	25
GET CURRENT KEY	30 — 35
IF NO UFO ON SCREEN GENERATE ONE	40 — 55
IF CRASH — END	60 — 65
IF UFO — MOVE UFO	70 — 95
IF UFO HIT — GO DOWN	100
IF UFO CRASH	105 — 145
END	150
IF KEY — 'S' FIRE	155 — 171
IF SHELL FIRED – MOVE SHELL	175 — 190
NEXT LOOP (GO TO 25)	200
CLEAR SCREEN PRINT MESSAGE	1000 — 1030
DATA	10000 — 10055

VARIABLES

CR = −1: NO UFO
0: UFO
1: UFO — HIT
HI = HIGH SCORE
SC = SCORE
K = CURRENT KEY PRESSED
Y = PLANE POSITION
SH = SHELL FIRED OR NOT

U.F.O.

```
  0 POKE 55,255 : POKE 56,44
  1 FOR I = 0 TO 35 : READ A : POKE 14 * 1024 + I,A
     : NEXT
  2 IF PEEK ( 53270 ) AND 8 THEN POKE 53270,
     PEEK ( 53270 ) - 8
  3 FOR I = 0 TO 126 : READ A : POKE 832 + I,A :
     NEXT : V = 53248 : MC = 14 * 1024 : KB = 197
  4 FOR I = 0 TO 126 : READ A : POKE 15 * 1024 + I,A
     : NEXT
  6 FOR I = 0 TO 31 : READ A : POKE 12 * 1024 + I,A
     : NEXT : HI = 0
  7 POKE V + 32,6 : POKE V + 33,6 : POKE 54296,15
 10 POKE V + 37,3 : POKE V + 38,2 : SC = 0 :
     POKE V + 24,28
 15 POKE 2040,13 : POKE V + 39,5 : POKE V,100 :
     POKE V + 1,100 : POKE V + 21,1 : POKE V + 28,1
 17 POKE 2041,14 : POKE V + 40,5 : CR = - 1 :
     SH = - 1 : POKE 2042,240 : POKE V + 41,0
 20 PRINT "{CLR}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}
      {DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}@ABCC@A@@BBACCCBA@AC
     @ABBCA@CBBBCA@@ACCB";
 22 A = PEEK ( V + 30 )
 25 SYS MC : IF PEEK ( V + 22 ) = 199 THEN
     POKE 2022, RND ( 1 ) * 4
 30 K = PEEK ( KB )
 35 Y = PEEK ( V + 1 ) :
     Y = Y - 2 * ( K = 9 ) * ( Y > 45 ) +
     ( K = 23 ) * ( Y < 234 ) * 2
 36 IF Y > = 234 THEN 1000
 40 POKE V + 1,Y : IF CR > - 1 THEN 60
 45 IF RND ( 1 ) > .1 THEN 150
 50 POKE V + 21, PEEK ( V + 21 ) OR 2 :
     POKE V + 2,80 : POKE V + 16, PEEK ( V + 16 )
     OR 2
 55 POKE V + 3, INT ( RND ( 1 ) * 175 + 44 ) :
     CR = 0
 60 F = 0 : A = PEEK ( V + 2 ) :
     B = PEEK ( V + 16 ) AND 2 : A = A - 2 :
     IF A < 0 AND B THEN B = 0 : A = 254 : F = 1
 65 IF A < 0 AND CR = 0 THEN 1000
 67 IF A < 0 THEN CR = - 1 : GOTO 150
 70 POKE V + 2,A : POKE V + 16, PEEK ( V + 16 ) OR B
     : IF F THEN POKE V + 16, PEEK ( V + 16 ) - 2
 80 IF CR THEN 100
 85 A = PEEK ( V + 3 ) + INT ( RND ( 1 ) * 3 - 1 )
     * 3 : IF A < 44 THEN A = 44
 90 IF A > 235 THEN A = 235
 95 POKE V + 3,A : GOTO 150
100 POKE ( V + 3 ) , PEEK ( V + 3 ) + 2 :
     IF PEEK ( V + 3 ) < 235 THEN 150
105 POKE V + 28,3 : POKE V + 40,8 : POKE 2041,241
110 POKE 54276,0 : POKE 54277,15 : POKE 54273,72 :
     POKE 54272,169 : POKE 54276,129
115 FOR I = 1 TO 2000 : NEXT
```

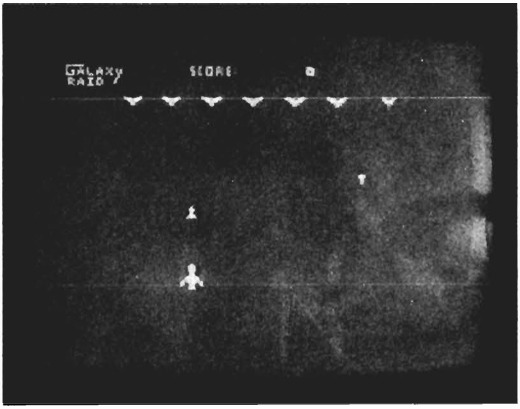
```
140 POKE V + 21, PEEK ( V + 21 ) - 2 : POKE V + 28,1
145 CR = - 1 : POKE V + 40,5 : POKE 2041,14
150 A = PEEK ( V + 30 ) : IF A = 3 THEN 1000
155 IF A = 6 AND CR = 0 THEN SH = - 1 : CR = 1 :
      POKE V + 21, PEEK ( V + 21 ) - 4 :
      A = PEEK ( V + 30 ) : SC = SC + 10
160 IF K < > 13 THEN 175
165 SH = 1 : POKE V + 21, PEEK ( V + 21 ) OR 4 :
      POKE V + 4, PEEK ( V ) + 10 : POKE V + 5,
      PEEK ( V + 1 )
170 IF PEEK ( V + 16 ) AND 4 THEN POKE V + 16,
      PEEK ( V + 16 ) - 4
171 POKE 54276,0 : POKE 54277,31 : POKE 54273,8 :
      POKE 54272,147 : POKE 54276,129
175 IF SH < 1 THEN 200
180 A = PEEK ( V + 4 ) : B = PEEK ( V + 16 ) AND 4
      : A = A + 3 : IF A > 255 THEN B = 4 : A = 1
185 IF A = 80 AND B THEN POKE V + 21,
      PEEK ( V + 21 ) - 4 : SH = - 1 :
      POKE 54273,0 : POKE 54272,0 : GOTO 200
190 POKE V + 4,A : POKE V + 16, PEEK ( V + 16 )
      OR B
200 GOTO 25
1000 POKE 2040,241 : POKE V + 39,8 : POKE 54273,72 :
      POKE 54272,169 : POKE 54277,47
1001 POKE 54276,0 : POKE 54276,129 :
      FOR I = 1 TO 3000 : NEXT : POKE 54273,0 :
      POKE 54272,0
1005 POKE V + 24,20 : POKE V + 21,0 :
      POKE V + 22,200 : PRINT " ♥ ▤YOU ▲ SCORED";SC
      : IF SC > HI THEN HI = SC
1010 PRINT "HI ▲ SCORE ▲ IS";HI : PRINT " [Q] [Q] [Q]WANT
       ▲ ANOTHER ▲ GAME?"
1020 GET A$ : IF A$ = "Y" THEN 10
1025 IF A$ < > "N" THEN 1020
1030 END
10000 DATA 174,22,208,202,224,191,240,4,142,22
      ,208,96,162,199,142,22,208
10005 DATA 162,0,189,193,7,157,192,7,232,224
      ,40,208,245,169,32,141,231,7,96
10010 DATA 0,0,0,0,0,0,0,0,0,0,0,0,128,0,0,128
      ,0,0,224,0,0,232,1,0,234,149,64
10015 DATA 170,170,168,63,170,255,22,169,80,10
      ,160,0,42,148,0,42,0,0,0,0,0,0,0,0
10020 DATA 0,0,0,0,0,0,0,0,0,0,0,0,99
10025 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
      ,0,0,0,0,0,0,0,0,0,0,0,0,24,0,0,255,0
10030 DATA 7,255,224,9,36,144,15,255,240,1
      ,255,128,0,126,0,0,0,0,0,0,0,0,0,0
10035 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
      ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
10040 DATA 0,0,0,0,0,0,0,0,194,0,0,63,128,0
      ,194,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
10042 DATA 0,0,0,0,99,0,0,0,0,12,0,0,63,0,0
      ,59,0,0,59,0,252,59,0,59,251,240
```

```
10044 DATA 58,171,176,14,170,255,3,170,171
       ,0,250,172,0,234,240,0,59,0,0,59,0
10046 DATA 0,238,192,3,174,192,14,190,192,14
       ,206,192,59,3,0,0,0,0,0,0,0
10050 DATA 0,0,0,0,0,98,247,255,0
       ,0,0,4,14,31,127,255
10055 DATA 0,0,16,24,60,124,254
       ,255,0,0,0,0,4,46,127,255
```

CHEXSUM

```
  0=980
  1=2350
  2=2477
  3=4336
  4=2437
  6=2845
  7=2005
 10=2583
 15=4064
 17=3984
 20=5313
 22=895
 25=2776
 30=688
 35=4527
 38=1165
 40=1677
 45=1111
 50=3820
 55=2253
 60=5498
 65=1430
 67=1791
 70=4255
 80=618
 85=3571
 90=1154
 95=1200
100=3079
105=1985
110=3378
115=1020
140=2182
145=1852
150=1797
155=5940
160=1032
165=4584
170=2758
171=3322
175=868
180=4333
185=5041
190=2189
200=473
1000=3416
1001=3759
1005=5838
1010=3339
1020=1243
1025=1184
1030=129
10000=3563
10005=3868
10010=4003
10015=4399
10020=1533
10025=4212
10030=4186
10035=3861
10040=4211
10042=3812
10044=3955
10046=3492
10050=2424
10055=2611     TOTAL= 136684
```

Galaxy Raid



A fleet of aliens looms above you, one by one they dive to attack you with their guided missiles.

You must avoid their missiles and destroy the alien ships. The main group is protected by a force field and only diving craft are vulnerable to your counter-fire. If you destroy the fleet another appears to replace it.

Your controls are:
- 'Z' for left
- 'C' for right
- 'M' to fire a missile

PROGRAM STRUCTURE	LINES
INITIALIZATION	0 — 9
PUT RAIDERS	10, 612, 614, 700 — 720
DISPLAY SCORE	500
GET KEYBOARD CHARACTER	510
MOVE ACCORDING TO KEY	520 — 530
FIRE IF (SPACE BAR) PRESSED	535, 900 — 905
TEST IF PLAYER HIT	545
TURN OFF SOUND	546
MOVE PLAYER'S MISSILE UP SCREEN	570
CHECK IF PLAYER'S AND RAIDER'S MISSILES COLLIDE	571
UPDATE SCORE	575
ELIMINATE PLAYER'S MISSILE IF PAST RANGE	580
HOME RAIDER'S MISSILE TOWARD PLAYER	582 — 584
PLAY MUSIC BETWEEN GAMES	800 — 850
CHECK IF RAIDER'S MISSILE HIT PLAYER	1000 — 1040
DATA	9000 — 10070

VARIABLES
S = SCORE
SP = NUMBER OF RAIDERS LEFT
SC = SCREEN REFERENCE POINT
X1 = HORIZONTAL POSITION OF PLAYER'S MISSILE
Y1 = VERTICAL POSITION OF PLAYER'S MISSILE
PE = VALUE PEEKED FROM KEYBOARD
P = HORIZONTAL POSITION OF PLAYER
I = LEFT HAND END OF RAIDER FORMATION

```
  0 PRINT "♥ ⊓FGHIJK ▲ ▲ ▲ ▲ ▲ ▲ ▲ ■PQRST" :
      POKE 53280,0 : POKE 53281,0 : V = 53248
  1 PRINT "◙ ◘ ⊓MINO ▲L ■" : POKE V + 24,28 :
      FOR I = 12288 TO 12463 : READ Q : POKE I,Q :
      NEXT
  2 FOR I = 12544 TO 12551 : POKE I,0 : NEXT
  3 POKE 54273,0 : FOR I = 12672 TO 12751 : READ Q :
      POKE I,Q : NEXT
  4 POKE V + 21,7 : POKE 2040,208 : POKE 2041,209 :
      POKE V + 39,8 : POKE V + 40,7 : POKE 2042,210
  5 POKE V + 41,2 : FOR I = 0 TO 190 : READ Q :
      POKE 13312 + I,Q : NEXT
  6 P = 120 : S = 0 : SP = 4 : AL = 7 : CO = 55296 :
      SC = 1024 : POKE V,P : POKE V + 1,200
  7 X = 5 + INT ( RND ( 1 ) * 20 ) : Y = 120 :
      X2 = X : Y2 = 160
  8 GOSUB 800 : POKE 54277,0 : POKE 54278,240 :
      GOSUB 700
  9 POKE V + 28,4 : POKE V + 38,5 :
      PE = PEEK ( V + 31 )
 10 FOR I = 1 TO 18 : PRINT "◙ ◘ ◘ ◘";
      TAB( I - 1 ) ;C$ : PRINT "◙ ◘ ◘ ◘"; TAB( I )
      ;A$
 20 GOSUB 500
 60 PRINT "◙ ◘ ◘ ◘"; TAB( I ) ;B$
 70 GOSUB 500
 75 NEXT
100 IF SP > 1 THEN SP = SP - 1
110 FOR I = 17 TO 1 STEP - 1 : PRINT "◙ ◘ ◘ ◘";
      TAB( I + 1 ) ;C$ : PRINT "◙ ◘ ◘ ◘"; TAB( I )
      ;A$
120 GOSUB 500
160 PRINT "◙ ◘ ◘ ◘"; TAB( I ) ;B$
170 GOSUB 500
180 NEXT
190 GOTO 10
500 PRINT "◙ ▣ ▣ ▣ ▣ ▣ ▣ ▣ ▣ ▣ ▣ ▣ ▣ ▣ ▣ ▣
      ▣ ▣ ▣ ▣ ▣ ▣ ▣ ▣";S;" ■"
502 IF SP < 2 THEN PRINT ""
505 FOR J = 1 TO INT ( SP )
510 PE = PEEK ( 197 )
520 IF PE = 12 AND P > 35 THEN P = P - 4
530 IF PE = 20 AND P < 320 THEN P = P + 4
535 IF Y1 = 0 AND PE = 36 THEN GOSUB 900
537 IF P > 255 THEN POKE V,P - 256 : POKE V + 16,
      PEEK ( V + 16 ) OR 1 : GOTO 545
540 POKE V,P : POKE V + 16, PEEK ( V + 16 ) AND 254
545 IF PEEK ( V + 31 ) = 1 THEN 1000
546 POKE 54273,0
560 IF Y1 = 0 THEN 581
570 Y1 = Y1 - 6 : POKE V + 3,Y1
571 IF PEEK ( V + 31 ) < > 2 THEN 580
572 POKE 54296,15 : POKE 54276,0 : POKE 54276,129
573 IF PEEK ( V + 16 ) AND 2 THEN POKE V + 16,
```

```
     PEEK ( V + 16 ) OR 4 : GOTO 575
574 POKE V + 16, PEEK ( V + 16 ) AND 251
575 POKE V + 4, PEEK ( V + 2 ) : POKE V + 5,Y1 :
     Y1 = 0 : S = S + 30 - Y / 40
576 POKE SC + X + Y,32 : POKE SC + X + Y + 1,32 :
     Y = 840
578 FOR Z = 40 TO 1 STEP - .6 : POKE 54273,Z : NEXT
     : POKE 54273,0 : POKE V + 4,0 : POKE V + 5,0
580 IF Y1 < 90 THEN Y1 = 0 : POKE V + 2,0 :
     POKE V + 3,0
581 POKE SC + X2 + Y2,32
582 IF RND ( 1 ) < .1 THEN X2 = X2 +
     ( X2 > ( P - 20 ) / 8 ) -
     ( X2 < ( P - 20 ) / 8 )
583 Y2 = Y2 + 40 : IF Y2 > 840 THEN Y2 = Y + 40 :
     X2 = X
584 POKE CO + X2 + Y2,3 : POKE SC + X2 + Y2,21
600 NEXT : DX = X : DY = Y
601 IF DY < 680 OR RND ( 1 ) < .6 THEN DY = DY + 40
602 IF DY < 360 AND DX > 20 AND DX < 38 THEN
     DX = DX + 1
604 IF DY < 360 AND DX < 21 AND DX > 1 THEN
     DX = DX - 1
605 IF DY > 400 AND DX > 20 AND DX > 1 THEN
     DX = DX - 1
606 IF DY > 400 AND DX < 21 AND DX < 38 THEN
     DX = DX + 1
608 IF DY < 840 THEN 620
609 AL = AL - 1 : IF AL = 0 THEN AL = 6 : DY = 120
     : DX = I + INT ( RND ( 1 ) * 6 ) * 4 :
     SP = SP - 1 : GOSUB 700
610 DX = INT ( RND ( 1 ) * 6 ) * 4 + 1 :
     IF MID$ ( A$,DX,2 ) = "▲▲" THEN 610
612 A1$ = LEFT$ ( A$,DX - 1 ) + "▲▲" + RIGHT$
     ( A$,24 - DX - 1 ) : A$ = A1$
614 A1$ = LEFT$ ( B$,DX - 1 ) + "▲▲" + RIGHT$
     ( B$,24 - DX - 1 ) : B$ = A1$
615 POKE 54296,5 : POKE 54276,0 : POKE 54276,17 :
     POKE 54273,50 : FOR Z = 1 TO 100 : NEXT
616 POKE 54273,0
619 DY = 120 : DX = I + DX - 1
620 POKE SC + Y + X,32 : POKE SC + Y + X + 1,32
625 POKE SC + DY + DX,4 : POKE SC + DY + DX + 1,5 :
     POKE CO + DY + DX,4 : POKE CO + DY + DX + 1,4
626 X = DX : Y = DY
630 RETURN
700 POKE 54296,8 : POKE 54276,0 : POKE 54276,33
705 A$ = "@A▲▲@A▲▲@A▲▲@A▲▲@A▲▲@A▲▲" :
     B$ = "BC▲▲BC▲▲BC▲▲BC▲▲BC▲▲BC▲▲"
710 C$ = "▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲▲
     ▲▲▲▲"
712 FOR Z = 1 TO 20 : POKE 54273,Z : NEXT :
     POKE 54273,0
720 RETURN
800 POKE 54276,0 : POKE 54277,0 : POKE 54278,0
802 POKE 54296,15 : POKE 54277,190 : POKE 54278,64 :
```

```
     POKE 54276,17
 805 FOR I = 1 TO 10 STEP .05 : POKE 54273,I : NEXT
 810 FOR I = 1 TO 30 STEP .2 : POKE 54273,I : NEXT
 815 FOR I = 50 TO 20 STEP - .2 : POKE 54273,I :
     NEXT
 820 FOR I = 80 TO 140 STEP 2.5 : POKE 54273,I :
     NEXT
 825 FOR I = 200 TO 50 STEP - 3 : POKE 54273,I :
     NEXT
 830 FOR I = 10 TO 40 STEP .2 : POKE 54273,I : NEXT
 835 FOR I = 80 TO 140 STEP 2 : POKE 54273,I : NEXT
 840 FOR I = 10 TO 1 STEP - .04 : POKE 54273,I : NEXT
 842 FOR I = 1 TO 10 STEP .05 : POKE 54273,I : NEXT
 845 FOR I = 100 TO 70 STEP - 2 : POKE 54273,I : NEXT
 850 POKE 54273,0 : RETURN
 900 POKE 54296,10 : POKE 54276,0 : POKE 54276,129
 901 IF P > 255 THEN POKE V + 2,P - 256 :
     POKE V + 16, PEEK ( V + 16 ) OR 2 : GOTO 905
 902 POKE V + 2,P : POKE V + 16, PEEK ( V + 16 )
     AND 253
 905 Y1 = 180 : FOR Z = 1 TO 30 STEP 2 :
     POKE 54273,Z : NEXT : RETURN
1000 POKE V,0 : POKE V + 1,0 : POKE V + 2,0 :
     POKE V + 3,0 : POKE V + 5,200
1005 IF P > 255 THEN POKE V + 4,P - 256 :
     POKE V + 16, PEEK ( V + 16 ) OR 4 : GOTO 1010
1007 POKE V + 4,P : POKE V + 16, PEEK ( V + 16 )
     AND 251
1010 POKE 54296,15 : POKE 54276,0 : POKE 54276,129
1020 FOR I = 1 TO 100 : POKE 54273,I : NEXT
1030 FOR I = 100 TO 50 STEP - 2 : POKE 54273,I : NEXT
1032 FOR I = 80 TO 1 STEP - .4 : POKE 54273,I :
     NEXT : POKE 54273,0
1035 POKE V + 4,0 : POKE V + 5,0 : POKE V + 29,0
1040 RUN
9000 DATA 96,125,31,15,2,1,0,0,3,31,124,248
     ,160,192,128,128
9010 DATA 0,0,127,31,6,1,0,0,0,0,127
     ,252,176,192,128,128
9020 DATA 28,14,7,3,2,1,0,0,28,56,112
     ,224,160,192,128,128
9030 DATA 255,192,192,192,207,195,195,255,255
     ,0,12,12,51,63,51,51
9040 DATA 0,0,48,48,48,48,48,63,0,0,12,12,51,63,51,51
9050 DATA 0,0,51,51,12,12,51,51,0
9060 DATA 0,0,51,51,27,27,12,12,12,48,48,96,96,192,0
9070 DATA 0,0,60,51,51,60,54,51,0,0,63,12,12,12,12,63
9075 DATA 0,0,60,51,51,51,51,60
9080 DATA 0,243,195,195,243,51,51,243,0,207
     ,12,12,12,12,12,207
9090 DATA 0,207,204,204,207,204,204,204,0,15,204
     ,204,15,204,204,207
9095 DATA 0,0,48,0,0,48,0,0,0,24,126,60,24,24,24,24
9100 DATA 0,120,126,102,102,102,126,0,0,24
     ,120,24,28,28,126,0
9110 DATA 0,126,6,24,96,102,126,0,0
```

```
       ,30,126,6,62,6,126,0
 9120 DATA 0,96,108,108,126,28,28,0,0
       ,14,126,96,24,6,126,0
 9130 DATA 0,126,112,126,102,102,126,0,0,126
       ,102,12,24,48,48,0
 9140 DATA 0,126,102,24,102,102,126,0,0,126
       ,102,102,126,14,14,0
10000 DATA 0,12,0,0,30,0,0,63,0,0,63,0,0,63,0
       ,0,63,0,0,30,0,0,12,0,0,30,0,0,63,0
10010 DATA 8,255,196,9,255,228,9,255,228,11
       ,127,180,15,127,188,14,63,28,12,12,12
10020 DATA 8,30,4,0,63,0,0,63,0,0,0,0,99
10030 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
       ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
10040 DATA 0,0,0,12,0,0,12,0,0,30,0,0,30,0,0
       ,12,0,0,12,0,0,30,0,0,63,0,0,109,128
10050 DATA 99,0,0,0,0,8,0,2,8,8,0,8,32,0,42,128,0
       ,174,160,2,239,128,42,255,160
10060 DATA 11,255,224,2,255,224,2,255,128,2,190
       ,0,8,186,128,0,184,32,0,40,0
10070 DATA 0,32,0,32,32,0,0,0,128,2,0,2,0,0,0,0,0,0
```

CHEXSUM

```
  0=3689     505=1053    601=2489
  1=4141     510=796     602=2901
  2=1766     520=1344    604=2832
  3=2732     530=1992    605=2828
  4=4129     535=1654    606=2897
  5=2929     537=3742    608=1008
  6=4473     540=2040    609=5634
  7=2947     545=1456    610=3410
  8=2041     546=528     612=3614
  9=2471     560=851     614=3614
 10=2983     570=1443    615=3699
 20=299      571=1582    616=528
 60=846      572=1966    619=1559
 70=299      573=3412    620=2342
 75=131      574=1572    625=5514
100=1486     575=3634    626=936
110=3464     576=2855    630=143
120=299      578=4101    700=1837
160=846      580=2447    705=4304
170=299      581=1112    710=1549
180=131      582=4125    712=2213
190=472      583=2856    720=143
500=1461     584=2275    800=1765
502=954      600=1134    802=2710
```

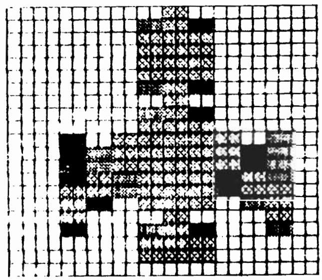
805=1920
810=1861
815=2075
820=2031
825=2094
830=1912
835=1936
840=2075
842=1920
845=2094
850=729
900=1962
901=3953
902=2298
905=2612
1000=3056
1005=4007
1007=2294
1010=1966
1020=1651
1030=2092
1032=2693
1035=1930
1040=139
9000=2860
9010=2692
9020=2738
9030=3204
9040=2522
9050=1383
9060=2455
9070=2497
9075=1261
9080=2986
9090=3286
9095=2363
9100=2929
9110=2602
9120=2746
9130=2955
9140=3008

10000=4367
10010=4457
10020=1738
10030=4088
10040=4369
10050=4217
10060=3818
10070=2311

TOTAL= 281468

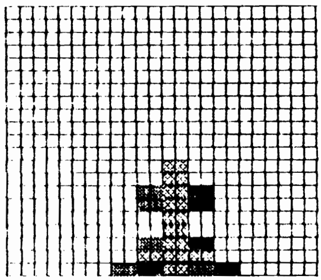
SPRITE

PLAYER'S SHIP



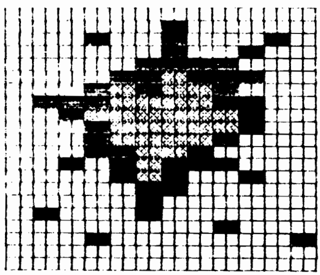
SPRITE

PLAYER'S MISSILE

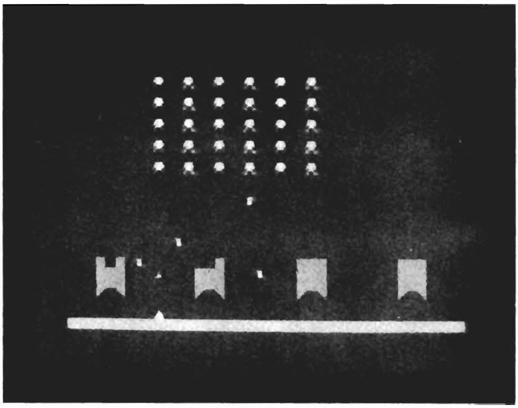


SPRITE

EXPLOSION



Invaders



Battle the invaders with your base on the ground. The arcade classic should have you addicted in no time at all.

Controls are: 'Z' left, 'C' right and 'M' to fire.

This program contains a large amount of machine code to control the invaders. Even if you don't as yet fully understand what is being done, don't worry. Just key the lines in carefully — especially the data lines. Check them thoroughly before trying to play the game.

PROGRAM STRUCTURE	LINES
INITIALIZE	0 — 2
ALTER MEMORY LIMIT	5
TEST IF 'Z' OR 'C' KEY PRESSED	100
MOVE LEFT/RIGHT	105 — 115
FIRE SHELL/SOUND/MOVE	200
SHELL HIT ANYTHING?	210
MOVE INVADERS	300
COLLISION WITH BARRIERS	315 — 325
MOVE BOMBS FROM ALIENS	500
CALL TO MACHINE CODE ROUTINE TO MANIPULATE INVADERS	1100
INVADERS' SOUND	1110 — 1115
END OF GAME?	1118
PRINT SCORE	1125

VARIABLES

V = START ADDRESS OF VIDEO CHIP
SC = SCORE
X, Y = COORDINATES OF BLASTER
K = CONTAINS KEY PRESSED ON KEYBOARD
P = CONTAINS THE CHARACTER ON THE SCREEN THAT THE SHELL HAS MADE CONTACT WITH (32 = SPACE)
SX = X COORDINATE OF SHELL
SY = Y COORDINATE OF SHELL
I = TEMPORARY VARIABLE
S(J) — ARRAY CONTAINING POSITIONS (6) OF ALIENS

INVADERS

```
  0 V = 53248 : POKE 54276,0 : POKE 54290,0 :
      POKE 54278,240 : POKE 54292,240 :
      POKE 54276,129
  1 POKE 54290,17 : POKE V + 33,0 : POKE V + 32,0 :
      SC = 1024 : CL = 55296 : CO = 54272 :
      POKE 54296,15
  5 POKE 56,28 : POKE 55,0 : FOR I = 12544 TO 12551
      : POKE I,0 : NEXT
 10 GOSUB 8000 : GOSUB 8060 : GOTO 1000
 99 IF K = 1 THEN 99
100 IF K < > 12 AND K < > 20 THEN 150
105 POKE SC + Y * 40 + X,32 : X = X - ( K = 12 ) *
      ( X > 0 ) + ( K = 20 ) * ( X < 39 ) :
      IF PEEK ( SC + Y * 40 + X ) < > 32 THEN D = 1
110 POKE CL + Y * 40 + X,3 : POKE SC + Y * 40 + X,0
115 RETURN
150 IF K = 36 AND SX < 0 THEN SX = X : SY = Y - 1 :
      POKE 54273,72 : POKE 54272,169
155 RETURN
200 POKE SC + SY * 40 + SX,32 : SY = SY - 1 :
      IF SY < 1 THEN SX = - 1 : RETURN
205 P = PEEK ( SC + SY * 40 + SX )
210 POKE CL + SY * 40 + SX,7 : POKE SC + SY * 40
      + SX,2 : IF P = 32 THEN RETURN
215 POKE 54273,34 : POKE 54272,75 : POKE SC + SY *
      40 + SX,4
220 IF P < > 1 THEN 250
225 FOR Y1 = 8 TO 0 STEP - 2 :
      FOR X1 = 10 TO 0 STEP - 2 :
      P = IN + Y1 * 6 + X1
230 IF PEEK ( P ) < > SY OR PEEK ( P + 1 ) < > SX
      THEN 245
235 S = S + ( 12 - Y1 ) * 5 : NO = NO - 1 :
      POKE P + 1,255 : X1 = 0 : Y1 = 0
245 NEXT : NEXT : GOTO 275
250 IF P < > 3 THEN 275
255 FOR J = 0 TO 5 : IF S ( J ) = SC + SY * 40 + SX
      THEN S = S + 5 : S ( J ) = 0 : J = 5
260 NEXT
275 POKE SC + SY * 40 + SX,32 : SX = - 1 : RETURN
300 POKE S ( J ) ,32 : S ( J ) = S ( J ) + 40 :
      IF S ( J ) > 1983 THEN S ( J ) = 0 : RETURN
305 P = PEEK ( S ( J ) ) : POKE S ( J ) + CO,3 :
      POKE S ( J ) ,3
310 IF P = 32 THEN RETURN
315 POKE S ( J ) ,4 : POKE 54273,43 : POKE 54272,52
      : IF P = 0 THEN D = 1
320 IF S ( J ) = SC + SX + SY * 40 THEN SX = - 1
325 POKE S ( J ) ,32 : S ( J ) = 0 : RETURN
500 S1 = INT ( RND ( 1 ) * 6 ) : IF S ( S1 ) > 0
      THEN RETURN
505 FOR Y1 = 8 TO 0 STEP - 2 : IF PEEK
      ( IN + Y1 * 6 + S1 * 2 + 1 ) = 255 THEN 550
510 S ( S1 ) = SC + ( PEEK
```

```
        ( IN + Y1 * 6 + S1 * 2 ) + 1 ) * 40 + PEEK
        ( IN + Y1 * 6 + S1 * 2 + 1 ) : Y1 = 0
 550 NEXT : RETURN
1000 D = 0 : FOR I = 1 TO SP : K = PEEK ( PE ) :
        IF K < > 64 THEN GOSUB 99
1005 IF SX > = 0 THEN GOSUB 200
1010 FOR J = 0 TO 5 : IF S ( J ) > 0 THEN GOSUB 300
1015 NEXT : POKE 54273,0 : POKE 54272,0 :
        POKE 54287,0 : POKE 54286,0
1020 IF RND ( 1 ) > .5 THEN GOSUB 500
1100 NEXT : SYS 7168 : IF NO < 21 THEN SP = 3 :
        IF NO < 11 THEN SP = 2 : IF NO < 6 THEN SP = 1
1101 POKE 54287,2 : POKE 54286,37 : IF
        PEEK ( 12303 ) = 36 THEN POKE 12303,129 :
        GOTO 1104
1102 POKE 12303,36
1104 IF PEEK ( 7604 ) = 0 AND D = 0 AND NO > 0
        THEN 1000
1105 IF NO = 0 THEN GOSUB 8050 : GOTO 1000
1110 POKE SC + Y * 40 + X,4 : POKE 54273,61 :
        POKE 54272,126
1115 FOR I = 15 TO 0 STEP - .1 : POKE 54296,I : NEXT
        : POKE 54287,0 : POKE 54286,0 :
        POKE SC + Y * 40 + X,0
1117 POKE 54273,0 : POKE 54272,0
1118 IF PEEK ( 7604 ) = 0 THEN B = B + 1 :
        IF B < 4 THEN POKE 54296,15 : GOTO 1000
1120 POKE V + 24,20 : PRINT "[CLR][Q][Q][Q][Q][Q][Q][Q][Q]
        [Q][Q][Q][Q][Q][Q][Q][Q][Q][Q][Q][Q][Q][Q][Q][Q]▓
        ▲▲▲▲▲▲▲▲▲▲▲▲▲**GAME▲OVER**"
1125 PRINT "[Q]YOU▲SCORED"S : RUN
8000 FOR AD = 7168 TO 7393
8005 READ A : POKE AD,A : NEXT
8010 FOR AD = 12288 TO 12351 : READ A : POKE AD,A :
        NEXT
8015 POKE V + 24,28
8050 FOR Y1 = 0 TO 4 : FOR X1 = 0 TO 5 :
        POKE 7620 + Y1 * 12 + X1 * 2,Y1 * 2 + 1 :
        POKE 7621 + Y1 * 12 + X1 * 2,X1 * 3
8052 NEXT : NEXT
8055 POKE 7605,1 : POKE 7604,0 : SX = - 1 : SY = 0 :
        NO = 30 : SP = 4 : RETURN
8060 B = 1 : PRINT "[CLR]" : FOR I = 1984 TO 2023
8065 POKE I,5 : POKE I + CO,6 : NEXT : X = 1 : Y = 23
        : PE = 197 : S = 0
8070 IN = 7620 : DIM S ( 5 ) : POKE CL + Y * 40 + X,3
        : POKE SC + Y * 40 + X,0
8075 FOR Y1 = 1 TO 4 : PRINT "[E][S][Q][Q][Q][Q][Q]
        [Q][Q][Q][Q][Q][Q][Q][Q][Q][Q][Q][Q][Q]" SPC( Y1 * 10 - 7 )
        "EEE[Q]▮▮▮▮▮▮EEE[Q]▮▮▮▮▮▮EEE";
8077 PRINT "[Q]▮▮▮▮▮▮F▲G";
8080 NEXT : RETURN
9000 DATA 162,7,142,179,29,162,0,142,180,29
        ,189,197,29,201,255,240,56,141,177,29
9001 DATA 189,196,29,141,176,29,169,32,141
        ,178,29,32,162,28,189,197,29,24,109
```

```
9005 DATA 181,29,157,197,29,201,0,208,5,160,1
     ,140,180,29,201,39,208,5,160,1,140
9010 DATA 180,29,141,177,29,169,1,141,178,29
     ,32,162,28,232,232,224,60,208,187
9015 DATA 173,180,29,208,1,96,169,0,141,180,29
     ,169,32,141,178,29,162,0,189
9020 DATA 197,29,201,255,240,27,141,177,29,189
     ,196,29,141,176,29,32,162,28,254
9025 DATA 196,29,189,196,29,201,22,208,5,160
     ,1,140,180,29,232,232,224,60,208
9028 DATA 216,173,181,29,201,1,240,4
     ,169,1,208,2,169,255
9030 DATA 141,181,29,173,180,29,208,3,76,0,28,96
9035 DATA 138,72,152,72,172,176,29,169,0,133,31
     ,169,4,133,32,162,0,24,169
9037 DATA 40,109,31,0,133,31,169,0,109,32,0,
     133,32,232,236,176,29
9040 DATA 208,235,172,177,29,173,178,29,145,31
     ,24,169,212,109,32,0,133
9045 DATA 32,173,179,29,145,31,104,168,104
     ,170,96,0,16,16,56,124
9050 DATA 124,254,254,126,90,255,189,189,36
     ,66,36,0,0,0,16,16,16,56,84,0,84,56
9055 DATA 16,56,56,16,0,108,213
     ,128,132,39,174,197,78
9060 DATA 255,255,255,255,255,255,255,255,255
     ,254,252,248,240,224,192,128
9065 DATA 255,127,63,31,15,7,3,1
```

CHEXSUM

0=4023	245=934	1020=1260
1=4890	250=982	1100=4497
5=2723	255=4336	1101=3928
10=1447	260=131	1102=585
99=739	275=2348	1104=2380
100=1773	300=3553	1105=1729
105=7265	305=2389	1110=2645
110=2596	310=820	1115=4791
115=143	315=2902	1117=1167
150=3937	320=2512	1118=3651
155=143	325=1345	1120=3974
200=4001	500=2283	1125=1348
205=1730	505=3910	8000=1089
210=3902	510=5670	8005=868
215=2848	550=332	8010=2246
220=974	1000=3269	8015=672
225=3304	1005=1215	8050=6147
230=2757	1010=1844	8052=320
235=3854	1015=2562	8055=3487

```
8060=1839
8065=3103
8070=3756
8075=4923
8077=970
8080=332
9000=4559
9001=4277
9005=4447
9010=4248
9015=3902
9020=4364
9025=4081
9028=2683
9030=2261
9035=3708
9037=3181
9040=3523
9045=3118
9050=4286
9055=2513
9060=3753
9065=1341

TOTAL= 218938
```

APPENDIX A

The purpose of this appendix is to provide you with information beyond just the program listings and stuctures.

The information presented here is an extract from our own programmers' notes.

Before we publish any games our programmers go through the programs in detail and in doing so come up with hints, tips and general information which would be useful to the programmer.

Be sure to progressively SAVE the programs onto cassette as you enter the listings. From time to time, the keyboard may 'lock up', expecially if you try to RUN a program that contains an error. We strongly recommend that you keep SAVEing copies of the program as you enter it into the computer and 'debug' any typing errors.

We also appreciate that some people like to 'customise' some of the games by enhancing them or changing them in some way. Anticipating this we have also suggested some changes you might like to make.

You will note that some of the programs contain machine code language subroutines. These are required to achieve effects that would be too slow in BASIC.

Don't worry if you are not familiar with machine language — the programs include all the instructions to initialize the machine language subroutine and to implement it. You must be especially careful with the DATA statments in these programs. Be sure to SAVE a copy of the program onto cassette before trying to RUN a program that contains machine language subroutines.

In Appendix D, we show the subroutines that make up the machine language subroutine in the program "U.F.O.". This subroutine scrolls the landscape at the bottom of the screen.

If there are any novices among you we have given a very detailed breakdown of "Start Pilot" under "Program Structure". You will be able to follow the program step-by-step and understand all it does.

A sprite generator with which to create sprites for your own games has been added. Use this to create good games to send to us for publishing together with the card in the back.

The true arcade games addict has been catered for with instructions on

how to convert the programs so that they are joystick compatable (in Appendix B).

All in all we feel that, whatever your purpose in purchasing this book, we have provided you with enough material to make your purchase more than worthwhile.

GENERAL HINTS

If you encounter difficulty trying to SAVE a program while in colour mode, hit RUN/STOP and RESTORE.

To get a completely black background and border : POKE 53280, 0 (border)
POKE 53281, 0 (background)
and CTRL 8 for white characters

If you crash the program during graphics mode and you can't read the line number in which you crashed, then type:
POKE 53272, 21 : POKE 53265, 155

If you type in a line that is exactly 2 screen lines long and can't get it accepted, use the cursor keys to move the cursor back into the line (at least one cursor left movement) and then save the line. If it still doesn't save, you have probably added redundant spaces or an extra character.

Some games use the space bar and others the 'M' key for firing. If you prefer one or the other you can alter as follows:
PEEK 197 checks last key pressed (returns an ASCII value)
When testing the value of last key pressed : 'M' = 36 and 'space bar' = 60

If you find that some games, that you know are correct, just don't work properly, try resetting and loading. We have found that previous games can sometimes corrupt pointers.

HINTS FOR SPECIFIC GAMES

PARANOID
This game takes about 30 seconds to initialize so don't assume you've made an error somewhere, just wait.

TURKEY
To speed up movement of the turkey change line 60 to read:
Y1 = Y1 + 2 * YD : X1 = X1 + 2 * XD
To speed up the waiter change lines 26 and 32 to P = P − 3 and P = P + 3 respectively.

HIGHER OR LOWER
Use the card selector and drawing routines as a basis for other card games.

FLIGHT SIMULATOR
To increase the rate of fuel consumption, and so force a faster landing and a time limit, decrease the 3200 in line 110. (A value of about 1500 seems to work well.)

SLALOM
To make the game more difficult, and thus more challenging, change line 30 to: 30 IF RND(1) < 1 etc.
This decreases the distance between flags and increases the number of trees. You could write a routine to incorporate different skill levels.

COWBOY SHOOTOUT
To change background and border colours change the vlaues POKEd into (V + 33) and *V + 32) in line 15.

DARK STAR
We encountered some problems loading Dark Star from cassette, presumably because of its length, if it happens just persevere. You might also like to change the sound on your shots to something more gutsy.

MARTIAN INVASION
Those of you who are observant will notice that at times a tripod will blow up before your bullet hits him — this is caused by a collision with the high X — bit.

UFO
To increase the speed of the game you can speed up the moving objects by making the following changes.

Line 35: To increase craft speed increase the value 2 in
Y=Y−2 and also the 2 at the very end of the line

Line 60: To increase UFO speed increase the value subtracted
from A in A=A−2

Line 180: To increase bullet speed increase the value added
to A in A=A+3

The UFO will not crash into the sea but should you desire it to do so just remove line 90.

APPENDIX B

TO CONVERT KEYBOARD GAMES TO JOYSTICK GAMES

— plug joystick into control port 1.

Where in the game you see PEEK (197) or a variable has been assigned to the value 197 (e.g. KB or PE in some cases) and you see PEEK (variable) then change the 197 to 56321. Using the above you would have PEEK (56321) or KB = 56321 or K = PEEK (56321).

The next step is to replace key tests with joystick value tests. Using the table below change the key tests as shown.

IF PE = "Fire key value"	IF PE AND 16 = 0
IF PE = "Up key value"	IF PE AND 1 = 0
IF PE = "Down key value"	IF PE AND 2 = 0
IF PE = "Left key value"	IF PE AND 4 = 0
IF PE = "Right key value"	IF PE AND 8 = 0

If you don't know the code values for keys and therefore don't know which key a value represents then run this little routine:

```
10 PRINT PEEK (197) ; : GOTO 10
```

RUN it and press the keys used in a game. The values for the keys will appear on the screen. Just relate these values according to the function of the key they represent to the joystick test values in the table.

e.g. North Sea Copter

```
40 PE = PEEK (56321)
45 IF PE AND 8 AND F > 0 THEN X = X + 2 : F = F -.1
   etc.
```

APPENDIX C

SPRITE GENERATOR Copyright (c) 1983 Beam Software

This program provides a fast and simple means for creating your own sprites.

The program provides a 24 x 21 grid on the screen on which to draw your sprites, the relevant key to be pressed being listed on the screen.

The real size sprite is shown on the screen, as well as the expanded (double resolution) sprite.

Keys used are:
A = MOVE CURSOR LEFT
D = MOVE CURSOR RIGHT
W = MOVE CURSOR UP
X = MOVE CURSOR DOWN
S = PLOT POINT ON GRID
SPACE BAR = ERASE POINT ON GRID
SHIFT CLR = CLEAR ENTIRE GRID
RETURN = CLEAR SCREEN AND PRINT OUT DATA

When return is pressed the sprite data is printed out, this may then be used in your own programs as follows:

```
 10 V = 53248 : REM START OF DISPLAY CHIP
 20 POKE V + 21, 1 : REM ENABLE SPRITE 0
 30 POKE 2040, 13 : REM SPRITE DATA POINTER
 40 POKE V + 39, 4 : REM COLOUR OF SPRITE
 50 FOR I = 0 TO 62 : READ A : POKE 832 + I, A : NEXT : REM
SPRITE DATA STORED HERE (13 * 64)
 60 FOR X = 0 TO 200 REM MOVE SPRITE (EXAMPLE)
 70 POKE V, X REM UPDATE X COORDINATE
 80 POKE V + 1, X REM UPDATE Y COORDINATE
 90 NEXT X
100 GOTO 60
200 REM DATA STATEMENTS WOULD FOLLOW
```

This program sets up the sprite and moves it accross the screen.

SPRITE GENERATOR Copyright (c) 1983 Beam Software

```
   1 SC = 1024
   2 POKE 649,0
  10 PRINT "▣"
  11 FOR I = 0 TO 7 : O ( 7 - I ) = 2 ↑ I : NEXT
  12 FOR I = 0 TO 7 : A ( I ) = 255 - O ( I ) : NEXT
  15 POKE 53281,6 : POKE 53280,6
  20 GOSUB 2000
  60 FOR I = 55296 TO 56319 : POKE I,0 : NEXT
 100 POKE 209,25 : POKE 210,4
 110 PRINT "▓SPACE ▴ ▴ ▴ ▴ ▥ERASE"
 120 POKE 209,65 : PRINT "▓S ▴ ▴ ▴ ▴ ▴ ▴ ▴ ▴ ▥PLOT"
 130 POKE 209,105 : PRINT "▓A ▴ ▴ ▴ ▴ ▴ ▴ ▴ ▴ ▥LEFT"
 140 POKE 209,145 : PRINT
       "▓D ▴ ▴ ▴ ▴ ▴ ▴ ▴ ▴ ▥RIGHT"
 150 POKE 209,185 : PRINT "▓W ▴ ▴ ▴ ▴ ▴ ▴ ▴ ▴ ▥UP"
 160 POKE 209,225 : PRINT "▓X ▴ ▴ ▴ ▴ ▴ ▴ ▴ ▴ ▥DOWN"
 170 POKE 209,9 : POKE 210,5 : PRINT
       "▓SHFT ▴CLR ▴ ▥CLEAR"
 180 POKE 209,49 : PRINT "▓RETURN ▴ ▴ ▴ ▥DATA"
 200 SO = 832 : POKE 2040,13 : POKE 53287,8
 201 POKE 2041,13 : POKE 53288,4 : POKE 53250,11 :
       POKE 53251,190
 205 GOSUB 3000
 210 POKE 53248,24
 220 POKE 53249,135
 225 POKE 53264,3
 230 POKE 53269,3
 240 POKE 53271,2 : POKE 53277,2
 300 CX = 0 : CY = 0
 500 FOR I = 0 TO 5
 505 FOR TP = 1 TO 20 : NEXT
 510 IF I = 0 THEN POKE SC + CX + CY * 40,
       PEEK ( SC + CX + CY * 40 ) OR 128
 520 IF I = 3 THEN POKE SC + CX + CY * 40,
       PEEK ( SC + CX + CY * 40 ) AND 127
 530 K = PEEK ( 197 )
 540 IF K = 10 THEN 5000
 550 IF K = 18 THEN 5100
 560 IF K = 13 THEN 5200
 570 IF K = 9 THEN 5300
 580 IF K = 23 THEN 5400
 590 IF K = 60 THEN 5500
 600 IF K = 51 THEN 5600
 610 IF K = 1 THEN 7000
 700 NEXT I
 800 GOTO 500
1000 GOTO 1000
2000 FOR I = 0 TO 20
2010 FOR J = 0 TO 23
2020 POKE SC + I * 40 + J,43
2030 NEXT J : NEXT I
2040 RETURN
3000 FOR I = 0 TO 62 : POKE SO + I,0 : NEXT : RETURN
5000 IF CX = 0 THEN 700
```

```
5010 POKE SC + CX + CY * 40,
       PEEK ( SC + CX + CY * 40 ) AND 127
5020 CX = CX - 1 : GOTO 700
5100 IF CX = 23 THEN 700
5110 POKE SC + CX + CY * 40,
       PEEK ( SC + CX + CY * 40 ) AND 127
5120 CX = CX + 1 : GOTO 700
5200 POKE SC + CX + CY * 40,81
5210 TP = SO + CY * 3 + INT ( CX / 8 )
5220 POKE TP, PEEK ( TP ) OR
       O ( CX - 8 * INT ( CX / 8 ) )
5230 GOTO 700
5300 IF CY = 0 THEN 700
5310 POKE SC + CX + CY * 40,
       PEEK ( SC + CX + CY * 40 ) AND 127
5320 CY = CY - 1 : GOTO 700
5400 IF CY = 20 THEN 700
5410 POKE SC + CX + CY * 40,
       PEEK ( SC + CX + CY * 40 ) AND 127
5420 CY = CY + 1 : GOTO 700
5500 POKE SC + CX + CY * 40,43
5510 TP = SO + 3 * CY + INT ( CX / 8 ) :
       POKE TP, PEEK ( TP ) AND
       A ( CX - 8 * INT ( CX / 8 ) )
5520 GOTO 700
5600 GOSUB 2000
5610 GOSUB 3000 : GOTO  700
7000 PRINT "♥"
7010 FOR I = 0 TO 62 : PRINT PEEK ( SO + I ) ;
7020 IF I < > 62 THEN PRINT ",";
7030 NEXT I
7040 END
```

CHEXSUM

```
1=544
2=420
10=372
11=1931
12=2061
15=1183
20=347
60=1783
100=968
110=1542
120=1948
130=2009
140=2031
150=1833
160=2018
170=2766
180=2148
200=1718
201=2539
205=348
210=592
220=653
225=526
230=535
240=1170
300=860
500=655
505=1007
510=3828
520=3824
530=704
540=892
550=901
560=896
570=844
580=899
590=898
600=899
610=836
700=206
800=529
1000=579
2000=709
2010=708
2020=1252
2030=475
2040=143
3000=1858
5000=867
5010=3217
5020=1349
5100=934
5110=3217
5120=1348
5200=1446
5210=1936
5220=2517
5230=529
5300=867
5310=3217
5320=1350
5400=933
5410=3217
5420=1349
5500=1446
5510=4642
5520=529
5600=347
5610=969
7000=372
7010=1690
7020=1198
7030=206
7040=129

TOTAL= 98267
```

APPENDIX D

ASSEMBLY LANGUAGE ROUTINE FROM U.F.O.

FINE SCROLLS SCREEN BY USING FINE SCROLL X REGISTER. WHEN THIS REGISTER HAS BEEN DECREMENTED BY 8 THE ENTIRE BOTTOM LINE IS COARSE SCROLLED.

```
3800  LDX $D016   ;READ SCROLL REGISTER
3803  DEX         ;DECREMENT THE VALUE
3804  CPX #191    ;IF DECREMENTED BY 8
                   ( 1 CHARACTER )
3806  BEQ $380C   ;THEN GOTO COARSE SCROLL SCREEN
3808  STX $D016
380B  RTS         ;RETURN
380C  LDX #199
380E  STX $D016   ;RESET SCROLL REGISTER
3811  LDX #0      ;SCROLL BOTTOM LINE OF SCREEN
3813  LDA $7C1,X
3816  STA $7C0,X
3819  INX
381A  CPX #40
381C  BNE $3813
381E  LDA #32    ;STORE NEW CHARACTER AT RIGHT
                  EDGE OF LINE
3820  STA $7E7
3823  RTS        ;RETURN
```

COMMODORE 64 GAMES BOOK

REGISTRATION CARD

Please fill out this page and return it promptly in order that we may keep you informed of new software and special offers that arise. Simply cut along the dotted line and return it to the correct address selected from those overleaf.

Where did you learn of this product?

☐ Magazine. If so, which one? ....

☐ Through a friend.

☐ Saw it in a Retail Store

☐ Other. Please specify ....

Which Magazines do you purchase?

Regularly: ....

Occassionally: ....

What age are you?

☐ 10-15 ☐ 16-19 ☐ 20-24 ☐ Over 25

We are continually writing new material and would appreciate receiving your comments on our product.

How would you rate this book?

☐ Excellent ☐ Value for money
☐ Good ☐ Priced right
☐ Poor ☐ Overpriced

Please tell us what software you would like to see produced for your COMMODORE 64

____

____

____

Name ____________________________________

Address ____________________________________

____________________________ Code __________

ED3

PUT THIS IN A STAMPED ENVELOPE AND SEND TO:

In the United States of America return page to:

Melbourne House Software Inc., 347 Reedwood Drive,
Nashville TN 37217.

In the United Kingdom return page to:

Melbourne House (Publishers) Ltd., Melbourne House, Church Yard,
Tring, Hertfordshire, HP23 5LU

In Australia & New Zealand return page to:

Melbourne House (Australia) Pty. Ltd., Suite 4, 75 Palmerston Crescent,
South Melbourne, Victoria, 3205.

The best software games book ever written for your Commodore 64.

These easy to enter program listings turn your Commodore 64 micro into an entire arcade of electronic fun and thrills. Experience action so fast and furious it will shatter all your concepts about computer game excitement.

Written by Clifford Ramshaw, author of the renowned VIC Innovative Computing and Mark Ramshaw of Zap Pow Boom VIC Arcade fame! These two programming wizards once again show how to get the ultimate game performance from your micro.

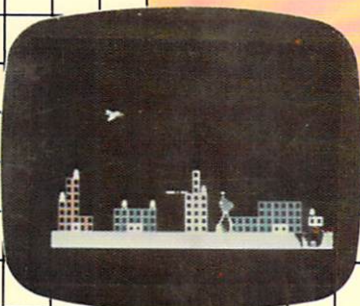
"An offering that definitely lives up to its title. The programs cover a range of games with a selection of shoot-em-up space wars, board games and a flight simulator." Personal Computer News.

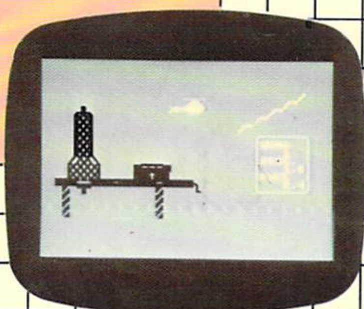
Every game program makes maximum use of the Commodore 64's sophisticated features, including its innovative Sprite capability and excellent sound.

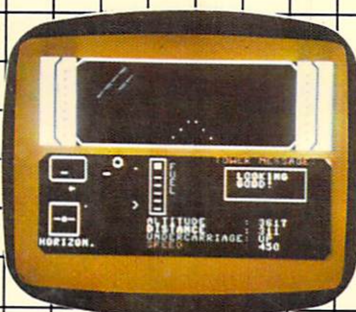
"For a Commodore 64 games enthusiast this book is a must." Personal Computer News.

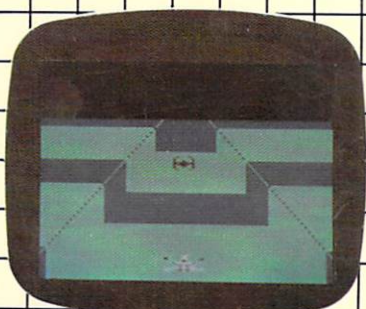
The Commodore 64 Games Book includes the unique CHEXSUM verification program that allows you to see at a glance if you have typed any program in this book correctly!

ISBN 0-86161-125-X
9 780861 611256









Melbourne House Publishers
ISBN 0 86161 125 X